PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H03K 19/177	A1	(11) International Publication Number: WO 95/28769
		(43) International Publication Date: 26 October 1995 (26.10.95)

(21) International Application Number: PCT/US95/04639

(22) International Filing Date: 14 April 1995 (14.04.95)

(30) Priority Data:
08/229,923 14 April 1994 (14.04.94) US

(71) Applicant: BTR, INC. [US/US]; c/o Corporate Trust Co., 1 East 1st Street, Reno, NV 89501 (US).

(72) Inventor: TING, Benjamin, S.; 21120 Sullivan Way, Saratoga, CA 95070 (US).

(74) Agents: SOBRINO, Maria, E. et al.; Blakely, Sokoloff, Taylor & Zafman, 7th floor, 12400 Wilshire Boulevard, Los Angeles, CA 90025-1026 (US).

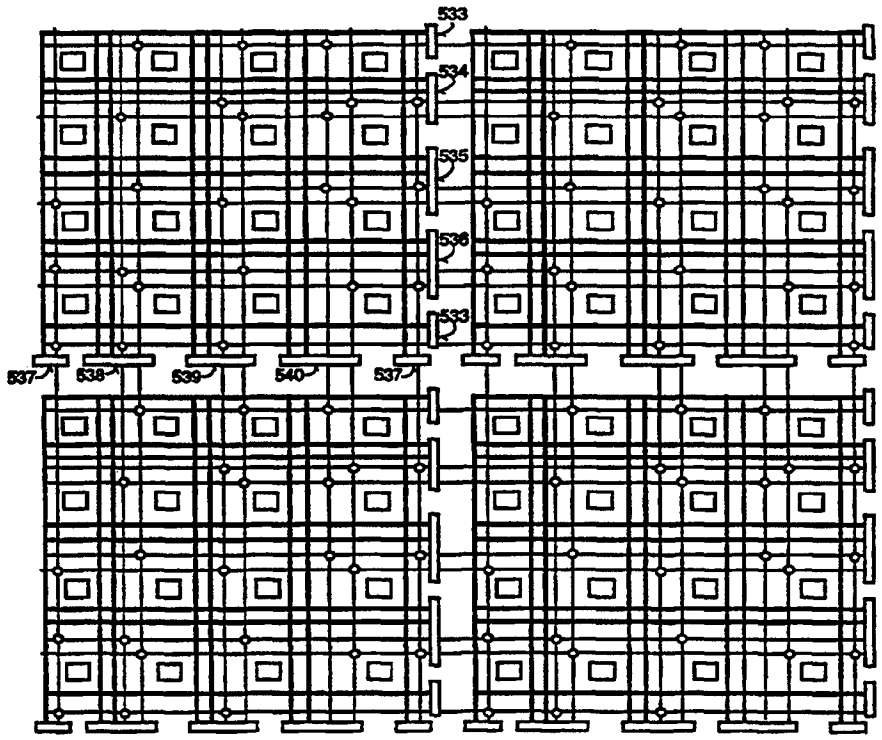
(81) Designated States: AM, AT, AT (Utility model), AU, BB, BG, BR, BY, CA, CH, CN, CZ, CZ (Utility model), DE, DE (Utility model), DK, DK (Utility model), EE, EE (Utility model), ES, FI, FI (Utility model), GB, GE, HU, IS, JP, KE, KG, KP, KR, KZ, LK, LR, LT, LU, LV, MD, MG, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SK (Utility model), TJ, TM, TT, UA, UG, UZ, VN, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG), ARIPO patent (KE, MW, SD, SZ, UG).

Published
With international search report.

(54) Title: ARCHITECTURE AND INTERCONNECT SCHEME FOR PROGRAMMABLE LOGIC CIRCUITS

(57) Abstract

An FPGA is comprised of a number of cells which perform logical functions on input signals. Programmable intraconnections provide connectability between each output of a cell belonging to a logical cluster to at least one input of each of the other cells belonging to that logical cluster. A set of programmable block connectors are used to provide connectability between logical clusters of cells and accessibility to the hierarchical routing network. A uniformly distributed first layer of routing network lines is used to provide connections amongst sets of block connectors. A uniformly distributed second layer of routing network lines is implemented to provide connectability between different first layers of routing network lines. Switching networks are used to provide connectability between the block connectors and routing network lines corresponding to the first layer. Other switching networks provide connectability between the routing network lines corresponding to the first layer to routing network lines corresponding to the second layer. Additional uniformly distributed layers of routing network lines are implemented to provide connectability between different prior layers of routing network lines.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	**GB**	United Kingdom	**MR**	Mauritania
AU	Australia	**GE**	Georgia	**MW**	Malawi
BB	Barbados	**GN**	Guinea	**NE**	Niger
BE	Belgium	**GR**	Greece	**NL**	Netherlands
BF	Burkina Faso	**HU**	Hungary	**NO**	Norway
BG	Bulgaria	**IE**	Ireland	**NZ**	New Zealand
BJ	Benin	**IT**	Italy	**PL**	Poland
BR	Brazil	**JP**	Japan	**PT**	Portugal
BY	Belarus	**KE**	Kenya	**RO**	Romania
CA	Canada	**KG**	Kyrgystan	**RU**	Russian Federation
CF	Central African Republic	**KP**	Democratic People's Republic of Korea	**SD**	Sudan
CG	Congo			**SE**	Sweden
CH	Switzerland	**KR**	Republic of Korea	**SI**	Slovenia
CI	Côte d'Ivoire	**KZ**	Kazakhstan	**SK**	Slovakia
CM	Cameroon	**LI**	Liechtenstein	**SN**	Senegal
CN	China	**LK**	Sri Lanka	**TD**	Chad
CS	Czechoslovakia	**LU**	Luxembourg	**TG**	Togo
CZ	Czech Republic	**LV**	Latvia	**TJ**	Tajikistan
DE	Germany	**MC**	Monaco	**TT**	Trinidad and Tobago
DK	Denmark	**MD**	Republic of Moldova	**UA**	Ukraine
ES	Spain	**MG**	Madagascar	**US**	United States of America
FI	Finland	**ML**	Mali	**UZ**	Uzbekistan
FR	France	**MN**	Mongolia	**VN**	Viet Nam
GA	Gabon				

ARCHITECTURE AND INTERCONNECT SCHEME FOR PROGRAMMABLE LOGIC CIRCUITS

CROSS-REFERENCE TO RELATED APPLICATIONS

This application is a continuation-in-part (CIP) application of Serial No. 08/101,197; filed August 3, 1993, which is assigned to the assignee of the present invention.

FIELD OF THE INVENTION

The present invention pertains to the field of programmable logic circuits. More particularly, the present invention relates to an architecture and interconnect scheme for programmable logic circuits.

BACKGROUND OF THE INVENTION

When integrated circuits (ICs) were first introduced, they were extremely expensive and were limited in their functionality. Rapid strides in semiconductor technology have vastly reduced the cost while simultaneously increased the performance of IC chips. However, the design, layout, and fabrication process for a dedicated, custom built IC remains quite costly. This is especially true for those instances where only a small quantity of a custom designed IC is to be manufactured. Moreover, the turn-around time (i.e., the time from initial design to a finished product) can frequently be

quite lengthy, especially for complex circuit designs. For electronic and computer products, it is critical to be the first to market. Furthermore, for custom ICs, it is rather difficult to effect changes to the initial design. It takes time, effort, and money to make any necessary changes.

In view of the shortcomings associated with custom IC's, field programmable gate arrays (FPGAs) offer an attractive solution in many instances. Basically, FPGAs are standard, high-density, off-the-shelf ICs which can be programmed by the user to a desired configuration. Circuit designers first define the desired logic functions, and the FPGA is programmed to process the input signals accordingly. Thereby, FPGA implementations can be designed, verified, and revised in a quick and efficient manner. Depending on the logic density requirements and production volumes, FPGAs are superior alternatives in terms of cost and time-to-market.

A typical FPGA essentially consists of an outer ring of I/O blocks surrounding an interior matrix of configurable logic blocks. The I/O blocks residing on the periphery of an FPGA are user programmable, such that each block can be programmed independently to be an input or an output and can also be tri-statable. Each logic block typically contains programmable combinatorial logic and storage registers. The combinatorial logic is used to perform boolean functions on its input variables. Often, the registers are loaded directly from a

logic block input, or they can be loaded from the combinatorial logic.

Interconnect resources occupy the channels between the rows and columns of the matrix of logic blocks and also between the logic blocks and the I/O blocks. These interconnect resources provide the flexibility to control the interconnection between two designated points on the chip. Usually, a metal network of lines run horizontally and vertically in the rows and columns between the logic blocks. Programmable switches connect the inputs and outputs of the logic blocks and I/O blocks to these metal lines. Crosspoint switches and interchanges at the intersections of rows and columns are used to switch signals from one line to another. Often, long lines are used to run the entire length and/or breadth of the chip.

The functions of the I/O blocks, logic blocks, and their respective interconnections are all programmable. Typically, these functions are controlled by a configuration program stored in an on-chip memory. The configuration program is loaded automatically from an external memory upon power-up, on command, or programmed by a microprocessor as part of system initialization.

The concept of FPGA was summarized in the sixty's by Minnick who described the concept of cell and cellular array as reconfigurable devices in the following documents:

Minnick, R.C. and Short, R.A., "Cellular Linear-Input Logic, Final Report," SRI Project 4122, Contract AF 19(628)-498, Stanford Research Institute, Menlo Park, California, AFCRL 64-6, DDC No. AD 433802 (February 1964); Minnick, R.C., "Cobweb Cellular Arrays," Proceedings AFIPS 1965 Fall Joint Computer Conference, Vol. 27, Part 1 pp. 327-341 (1965); Minnick, R.C. et al., "Cellular Logic, Final Report," SRI Project 5087, Contract AF 19(628)-4233, Stanford Research Institute, Menlo Park, California, AFCRL 66-613, (April 1966); and Minnick, R.C., "A Survey of Microcellular Research," Journal of the Association for Computing Machinery, Vol. 14, No. 2, pp. 203-241 (April 1967). In addition to memory based (e.g., RAM-based, fuse-based, or antifuse-based) means of enabling interconnects between devices, Minnick also discussed both direct connections between neighboring cells and use of busing as another routing technique. The article by Spandorfer, L. M., "Synthesis of Logic Function on an Array of Integrated Circuits," Stanford Research Institute, Menlo Park, Calif., Contract AF 19(628)2907, AFCRL 64-6, DDC No. AD 433802 (November 1965), discussed the use of complementary MOS bi-directional passgate as a means of switching between two interconnect lines that can be programmed through memory means and adjacent neighboring cell interconnections. In Wahlstrom, S. E., "Programmable Logic Arrays - Cheaper by the Millions," Electronics, Vol. 40, No. 25, 11, pp. 90-95

(December 1967), a RAM-based, reconfigurable logic array of a two-dimensional array of identical cells with both direct connections between adjacent cells and a network of data buses is described.

Shoup, R. G., "Programmable Cellular Logic Arrays," Ph.D. dissertation, Carnegie-Mellon University, Pittsburgh, PA (March 1970), discussed programmable cellular logic arrays and reiterates many of the same concepts and terminology of Minnick and recapitulates the array of Wahlstrom. In Shoup's thesis, the concept of neighbor connections extends from the simple 2-input 1-output nearest-neighbor connections to the 8-neighbor 2-way connections. Shoup further described use of bus as part of the interconnection structure to improve the power and flexibility of an array. Buses can be used to route signals over distances too long, or in inconvenient directions, for ordinary neighbor connections. This is particularly useful in passing inputs and outputs from outside the array to interior cells.

U.S. Patent Number 4,020,469 discussed a programmable logic array that can program, test, and repair itself. U.S. Patent Number 4,870,302 introduced a coarse grain architecture without use of neighbor direct interconnections where all the programmed connections are through the use of three different sets of buses in a channeled architecture. The coarse grain cell (called a Configurable Logical block or CLB) contains

both RAM-based logic table look up combinational logic and flip flops inside the CLB where a user defined logic must be mapped into the functions available inside the CLB. U.S. Patent Number 4,935,734 introduced a simple logic function cell defined as a NAND, NOR or similar types of simple logic function inside each cell. The interconnection scheme is through direct neighbor and directional bus connections. U.S. Patent Numbers 4,700,187 and 4,918,440 defined a more complex logic function cell where an Exclusive OR and AND functions and a register bit is available and selectable within the cell. The preferred connection scheme is through direct neighbor connections. Use of bi-direction buses as connections were also included.

Current FPGA technology has a few shortcomings. These problems are embodied by the low level of circuit utilization given the vast number of transistors available on chip provided by the manufacturers. Circuit utilization is influenced by three factors. The first one at the transistor or fine grain cell level is the function and flexibility of the basic logic element that can be readily used by the users. The second one is the ease in which to form meaningful macro logic functions using the first logic elements with minimum waste of circuit area. The last factor is the interconnections of those macro logic functions to implement chip level design efficiently. The fine grained cell architectures such as those

described above, provided easily usable and flexible logical functions for designers at the base logic element level.

However, for dense and complex macro functions and chip level routing, the interconnection resources required to connect a large number of signals from output of a cell to the input(s) of other cells can be quickly exhausted, and adding these resources can be very expensive in terms of silicon area. As a consequence, in fine grained architecture design, most of the cells are either left unused due to inaccessibility, or the cells are used as interconnect wires instead of logic. This adds greatly to routing delays in addition to low logic utilization, or excessive amount of routing resources are added, greatly increasing the circuit size. The coarse grain architecture coupled with extensive routing buses allows significant improvements for signals connecting outputs of a CLB to inputs of other CLBs. The utilization at the CLB interconnect level is high. However, the difficulty is the partitioning and mapping of complex logic functions so as to exactly fit into the CLBs. If a part of logic inside the CLB is left unused, then the utilization (effective number of gates per unit area used) inside the CLB can be low.

Another problem with prior art FPGAs is due to the fact that typically a fixed number of inputs and a fixed number of outputs are provided for each logic block. If, by happenstance,

all the outputs of a particular logic block is used up, then the rest of that logic block becomes useless.

Therefore, there is a need in prior art FPGAs for a new architecture that will maximize the utilization of an FPGA while minimizing any impact on the die size. The new architecture should provide flexibility in the lowest logic element level in terms of functionality and flexibility of use by users, high density per unit area functionality at the macro level where users can readily form complex logic functions with the base logic elements, and finally high percentage of interconnectability with a hierarchical, uniformly distributed routing network for signals connecting macros and base logic elements at the chip level. Furthermore, the new architecture should provide users with the flexibility of having the number of inputs and outputs for individual logical block be selectable and programmable, and a scalable architecture to accommodate a range of FPGA sizes.

SUMMARY OF THE INVENTION

The present invention relates to an architecture of logic and connection scheme for programmable logic circuits, such as those for field programmable gate arrays (FPGAs). The programmable logic circuit is comprised of a number of cells which perform digital functions on input signals. Depending on user's specific design, certain cells are programmably interconnected to a particular configuration for realizing the desired logic functions.

In the currently preferred embodiment, four logic cells (four two-input one-output logic gates and one D flip-flop) form a logical cluster (i.e. a 2x2 cell array) and four sets of clusters form a logical block (i.e. a 4x4 cell array). Within each cluster, there is a set of five intraconnection lines, called Intraconnection Matrix (I-Matrix), one associated with the output of each one of the four gates and the D flip-flop that is connectable to the input of the other cells. Within each logical block, the I-Matrix within each cluster can be extended to an adjacent cluster through a passgate to form connections within the logical block (to extend the intraconnection range). Inside each logical block, there is an associated set of access lines called Block Connectors (BCs). The block connectors provide access to and connectability between the various cells of that same logical block. In other words, each input and output of each of the cells of a logical block is capable of being connected

to a set of block connectors corresponding to that logical block. With the judicious use of I-Matrix and block connectors within the same logical block, a set of signals can be internally connected without using any resources outside the logical block. A number of programmable switches are used to control which of the block connectors are to be connected together to a set of inputs and/or outputs of the cells inside the logical block for external access connecting to signals outside the current logical block. In other words, the input and/or output pins inside a logical block that are to be externally connected outside of the current logical block are accessed or connected through block connectors within the current logical block.

In order to route signals between the various logical blocks, a uniformly distributed multiple level architecture (MLA) routing network is used to provide connectability between each of the individual sets of block connectors. Programmable switches are implemented to control which of the first level MLA routing network lines are to be connected together. Additional programmable switches are used to control which of the block connectors are to be connected to specific first level MLA routing lines. For example, the switches can be programmed to allow an originating cell belonging to one logical block to be connected to a destination cell belonging to a different logical block. This can be

accomplished by connecting the originating cell through one or more of its block connectors, onto the first level MLA, depending on the distance, other level(s) of MLA, and down through descending levels of MLAs back to the first level MLA, and finally through the block connector of the destination cell. Thereby, the block connectors and first level of MLA routing network provide interconnectability for an 8x8 cell array, called a block cluster.

In the present invention, larger cell arrays can be interconnected by implementing additional levels of MLA routing networks. For example, connectability for a 16x16 cell array, called a block sector, can be achieved by implementing a second level of MLA routing network lines to provide connectability between the various first level of MLA routing lines thereby making connections between different block clusters. Each level of MLA has a corresponding number of switches for providing programmable interconnections of the routing network of that level. Additional switching exchange networks are used to provide connectability between the various levels of MLAs.

In one embodiment, switches are used to provide connectability between two different sets of block connectors. Moreover, switches can be included to provide connectability between different sets of MLA routing lines of a particular level of MLAs. This provides for increased routing flexibility.

In the present invention, all MLA routing network lines are bi-directional. The switches are comprised of programmable bi-directional passgates. For increased number of levels, drivers may be necessary for providing the necessary switching speed for driving the routing lines, passgates, and associated loads, etc. In one embodiment, switches are used to provide programmable connectability amongst various sets of block connectors. Additional switches can be implemented to provide programmable connectability amongst various sets of the first level of MLA. This scheme can be repeated for higher levels of MLAs.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings and in which like reference numerals refer to similar elements and in which:

Figure 1 is a block diagram of a field programmable gate array logic upon which the present invention may be practiced.

Figure 2A shows one example of an individual cell.

Figure 2B shows another example of an individual cell.

Figure 3A shows a logical cluster.

Figure 3B shows the extension of I-matrix intraconnections of a logical cluster to a neighboring logical cluster.

Figure 4A shows an example of a logical cluster with vertical block connectors.

Figure 4B shows an example of a logical cluster with horizontal block connectors.

Figure 5A shows the eight block connector to level 1 MLA exchange networks associated with a logical block and level 1 MLA turn points.

Figure 5B shows a level 1 MLA turn point.

Figure 5C shows an exchange network.

Figure 6 shows the routing network for a block cluster.

Figure 7A shows the block diagram of a block sector.

Figure 7B shows a level 1 to level 2 MLA routing exchange network.

Figure 8A shows a sector cluster.

Figure 8B shows a level 2 to level 3 MLA routing exchange network.

Figure 9 shows one embodiment of a hierarchical multiple level routing network for providing routability between the logical blocks and the MLA levels.

Figure 10 shows another embodiment of a hierarchical multiple level routing network for providing routability between the logical blocks and the MLA levels.

Figure 11 shows a block diagram of one embodiment of the hierarchical routing network wherein two groups of block connectors access the same MLA lines.

Figure 12 shows a block diagram of part of the multiple level routing network which encompasses Block Connectors to the MLA-3 Level with MLA Tabs for higher levels of routing network.

Figure 13 shows an MLA-1 turn network.

Figure 14 shows an MLA-2 turn network.

Figure 15 shows an MLA-3 turn network.

Figure 16 shows one embodiment of a routing network for the MLA-4 layer and the mechanism whereby the MLA-4 lines are accessed.

Figure 17 shows three different switch embodiments.

Figure 18 shows one embodiment of a routing network for the MLA-5 layer and the mechanism whereby MLA-5 lines are accessed.

DETAILED DESCRIPTION

An architecture and interconnect scheme for programmable logic circuits is described. In the following description, for purposes of explanation, numerous specific details are set forth, such as combinational logic, cell configuration, numbers of cells, etc., in order to provide a thorough understanding of the present invention. It will be obvious, however, to one skilled in the art that the present invention may be practiced without these specific details. In other instances, well-known structures and devices are shown in block diagram form in order to avoid unnecessarily obscuring the present invention. It should also be noted that the present invention pertains to a variety of processes including but not limited to static random access memory (SRAM), dynamic random access memory (DRAM), fuse, anti-fuse, erasable programmable read only memory (EPROM), electrically erasable programmable read only memory (EEPROM), FLASH, and ferroelectric processes.

Referring to Figure 1, a block diagram of a field programmable gate array logic upon which the present invention may be practiced is shown as 100. The I/O logical blocks 102, 103, 111 and 112 provide an interface between external package pins of the FPGA and the internal user logic either directly or through the I/O to Core interface 104, 105, 113, and 114. Four interface blocks 104, 105, 113, and 114 provide decoupling

between core 106 and the I/O logic 102, 103, 111, and 112. Core 106 is comprised of a number of clusters 107 which are intraconnected by I-Matrix 101 and interconnected by MLA routing network 108.

Control/programming logic 109 is used to control all of the bits for programming the bit and word lines. For anti-fuse or fuse technology, high voltage/current is applied to either zap or connect a fuse. For EEPROM, Flash, or ferroelectric technology, there is an erase cycle followed by a programming cycle for programming the logic states of the memory bits. In order to minimize skewing, a separate clock/reset logic 110 is used to provide clock and reset lines on a group basis.

In the currently preferred embodiment, each of the clusters 107 is comprised of a 2x2 hierarchy of four cells, called a logical cluster. Figures 2A and 2B show examples of individual cells 200 and 250. Cell 200 performs multiple logic functions on two input signals (A and B) and provides an output signal X. In the currently preferred embodiment, cell 200 is comprised of an XOR gate 201, a two-input NAND gate 202, and a two-input NOR gate 203. It should be noted, however, that in other embodiments, cell 200 can include various other types and/or combinations of gates. Cell 250 is comprised of cell 200 coupled with a D flip flop cell 260. The output X of cell 200 can be programmed to connect directly to the data input D of the D flip flop gate 204 by activating

switch 218. The data input D can be accessed as a third input of the combined cell 250.

Each of the two input signals A and B and the D input of D flip-flop can be inverted or non-inverted, depending on the states of switches 206-211. Activating switches 206, 208 and 210 causes signals A, B and D to be driven by drivers 212-214 to gates 201-204 in a non-inverted fashion. Activating switches 207, 209, and 211 causes the input signals A, B and D to be inverted by inverters 215-217 before being passed to gates 201-204. The six switches 212-217 can individually be turned on and off as programmed by the user.

Note that the XOR gate 201, NAND gate 202, and NOR gate 203 can also be used to perform XNOR, AND and OR by propagating the output signal to the next stage, whereby the signal can be inverted as discussed above.

Three switches 219-221 are respectively coupled to the outputs of the three gates 201-203. Again, these switches are programmable by the user. Thereby, the user can specify which of the outputs from the gates 201-203 is to be sent to driver 224 as the output X from cell 200.

The aforementioned switches 206-211, 218-221 are comprised of bi-directional, program-controlled passgates. Depending on the state of the control signal, the switches are either conducting (i.e. passes a signal on the line) or non-conducting (i.e. does not pass the signal on the line). Switches

mentioned in the following sections are similarly comprised of program-controlled passgates.

Referring now to Figure 3A, a logical cluster 107 is shown. In the currently preferred embodiment, logical cluster 107 is comprised of four cells 301-304 and a D flip-flop 305, twenty five switches 306-330, and five intraconnection lines 331-335. The Intraconnection lines 331-335 and switches 306-330 form the I-Matrix. I-Matrix provide connectability of the output, X, of each of the four cells 301-304, and the output X of the D flip-flop 305 to at least one input of each of the other three cells and the D flip-flop. For example, the output X of cell 301 can be connected to input A of cell 302 by enabling switches 306 and 307. Likewise, the output X of cell 301 can be connected to input B of cell 303 by enabling switches 306 and 310. Output X of cell 301 can be connected to input A of cell 304 by enabling switches 306 and 308. Output X of cell 301 can be connected to input D of the D flip-flop cell 305 by enabling switches 306 and 309.

Similarly, the output X from cell 302 can be connected to input A of cell 301 by enabling switches 311 and 312. The output X from cell 302 can be connected to input A of cell 303 by enabling switches 311 and 315. The output X from cell 302 can be connected to input B of cell 304 by enabling switches 311 and 313. Output X of cell 302 can be connected to input D of the D flip-flop cell 305 by enabling switches 311 and 314.

Similarly, the output X from cell 303 can be connected to input B of cell 301 by enabling switches 326 and 327. The output X from cell 303 can be connected to input A of cell 302 by enabling switches 326 and 328. The output X from cell 303 can be connected to input B of cell 304 by enabling switches 326 and 329. Output X of cell 303 can be connected to input D of the D flip-flop cell 305 by enabling switches 326 and 330.

For cell 304, the output X from cell 304 can be connected to input B of cell 301 by enabling switches 316 and 317. The output X from cell 304 can be connected to input B of cell 302 by enabling switches 316 and 318. The output X from cell 304 can be connected to input A of cell 303 by enabling switches 316 and 319. Output X of cell 304 can be programmably connected to input D of the D flip-flop cell 305 by enabling switch 218 in Figure 2A.

With respect to cell 305, its output is connectable to the A input of cell 301 by enabling switches 320 and 321; the B input of cell 302 by enabling switches 320 and 322; the B input of cell 303 by enabling switches 320 and 325; the A input of cell 304 by enabling switches 320 and 323; and the D input of cell 305 itself by enabling switches 320 and 324.

It can be seen that each output of the cells 301-304 and of the D flip-flop 305 is connectable to the input of each of its neighboring cells and/or flip-flop inside the cluster.

In the currently preferred embodiment of the present invention, each logical cluster is connectable to all the other logical clusters inside each logical block through passgate switches extending the I-Matrix from neighboring clusters inside each logical block. Figure 3B illustrates the extension of I-Matrix intraconnection lines 331-335 of the cells 301-304 and the D flip-flop 305 of a logical cluster 107 to a neighboring logical cluster 107 through the passgate switches 336-355 within the same logical block.

In the currently preferred embodiment of the present invention, each logical block is connectable to all the other logical blocks of the FPGA. This is accomplished by implementing an architecture with multiple layers of interconnections. It is important to note that this multiple layers routing architecture is a conceptual hierarchy, not a process or technology hierarchy and is hence readily implementable with today's silicon process technology. The bottom most layer of interconnections is referred to as the "block connectors". A set of block connectors provides the access and interconnections of signals within an associated logical block (which is consisted of four logical clusters or 16 cells). Thereby, different sets of logical clusters within the same logical block are connectable to any of the other logical clusters within that group through the use of extended I-Matrix and/or block connectors. Again, programmable bi-

directional passgates are used as switches to provide routing flexibility to the user.

The next level of connections is referred to as the "level 1 Multiple Level Architecture (MLA)" routing network. The level 1 MLA routing network provides the interconnections between several sets of block connectors. Programmable passgates switches are used to provide users with the capability of selecting which of the block connectors are to be connected. Consequently, a first logical block from one set of logical block groups is connectable to a second logical block belonging to the same group. The appropriate switches are enabled to connect the block connectors of the first logical block to the routing lines of the level 1 MLA routing network. The appropriate switches of the level 1 MLA routing network are enabled to provide the connections to the block connectors of the second logical block to the routing lines of the level 1 MLA routing network. The appropriate switches are enabled to connect the routing lines of the level 1 MLA routing network that connected to the block connectors of the first and the second logical blocks. Furthermore, the user has the additional flexibility of programming the various switches within any given logical block to effect the desired intraconnections between each of the cells of any logical block.

The next level of connections is referred to as the "level 2 Multiple Level Architecture (MLA)" routing network. The

level 2 MLA provides the interconnections to the various level 1 MLA to effect access and connections of a block cluster. Again, bi-directional passgate switches are programmed by the user to effect the desired connections. By implementing level 2 MLA routing network, programmable interconnections between even larger numbers of logical blocks is achieved.

Additional levels of MLA routing networks can be implemented to provide programmable interconnections for ever increasing numbers and groups of logical blocks, block clusters, block sectors, etc. Basically, the present invention takes a three dimensional approach for implementing routing. Signals are routed amongst the intraconnections of a logical block. These signals can then be accessed through block connectors and routed according to the programmed connections of the block connectors. If needed, signals are "elevated" to the level 1 MLA, routed through the level 1 MLA routing network, "de-elevated" to the appropriate block connectors, and then passed to the destination logical block.

If level 2 MLA routing network is required, some of the signals are elevated a second time from a level 1 MLA routing network line or directly to the level 2 MLA routing network, routed to a different set of level 2 MLA routing network line, and "de-elevated" from the level 2 MLA routing network line to a Level 1 MLA routing network line. Thereupon, the signals are "de-elevated" a second time to pass the signal from the

level 1 MLA to the appropriate block connectors of the destination logical block. Alternatively, the "elevation" can be achieved directly without passing through the level 1 MLA routing network. This same approach is performed for level 3, 4, 5, etc. MLAs on an as needed basis, depending on the size and density of the FPGA. Partial level n MLA can be implemented using the above discussed method to implement a FPGA with a given cell array count.

Figure 4A shows an example of a logical cluster and the associated vertical block connectors within the logical block. In the currently preferred embodiment, each cell in a logical cluster is accessible from the input by two vertical block connectors and each output of the cell in a logical cluster is accessible to two of the vertical block connectors. For example, input A of cell 301 is accessible to the vertical block connectors 451 (BC-V11) and 453 (BC-V21) through switches 467, 462 respectively, input B of cell 301 is accessible to the vertical block connectors 452 (BC-V12) and 454 (BC-V22) through switches 466, 468 respectively, output X of cell 301 is accessible to the vertical block connectors 455 (BC-V31) and 458 (BC-V42) through switches 460, 459 respectively. Input A of cell 302 is accessible to the vertical block connectors 453 (BC-V21) and 455 (BC-V31) through switches 463, 464 respectively, input B of cell 302 is accessible to the vertical block connectors 454 (BC-V22) and 456 (BC-V32) through

switches 469, 470 respectively, output X of cell 302 is accessible to the vertical block connectors 452 (BC-V12) and 457 (BC-V41) through switches 461, 465 respectively. Input A of cell 303 is accessible to the vertical block connectors 451 (BC-V11) and 453 (BC-V21) through switches 485, 476 respectively, input B of cell 303 is accessible to the vertical block connectors 452 (BC-V12) and 454 (BC-V22) through switches 480, 476 respectively, output X of cell 303 is accessible to the vertical block connectors 455 (BC-V31) and 458 (BC-V42) through switches 472, 471 respectively. The input A of cell 304 is accessible to the vertical block connectors 453 (BC-V21) and 455 (BC-V31) through switches 477, 478 respectively, input B of cell 304 is accessible to the vertical block connectors 454 (BC-V22) and 456 (BC-V32) through switches 482, 484 respectively, output X of cell 304 is accessible to the vertical block connectors 452 (BC-V12) and 457 (BC-V41) through switches 475, 474 respectively. D flip-flop cell 305 input is accessible to the vertical block connectors 454 (BC-V22) and 455 (BC-V31) through switches 473, 479 respectively, output X of cell 305 is accessible to the vertical block connectors 452 (BC-V12) and 457 (BC-V41) through switches 483, 486 respectively.

In similar fashion, Figure 4B shows the possible connections corresponding to horizontal block connectors and the logical cluster shown in Figure 4A. Input A of cell 301 is

accessible to the horizontal block connectors 402 (BC-H12) and 404 (BC-H22) through switches 409, 413 respectively, input B of cell 301 is accessible to the horizontal block connectors 401 (BC-H11) and 403 (BC-H21) through switches 415, 416 respectively, output X of cell 301 is accessible to the horizontal block connectors 405 (BC-H31) and 408 (BC-H42) through switches 421, 428 respectively. Input A of cell 302 is accessible to the horizontal block connectors 402 (BC-H12) and 404 (BC-H22) through switches 411, 414 respectively, input B of cell 302 is accessible to the horizontal block connectors 401 (BC-H11) and 403 (BC-H21) through switches 433, 417 respectively, output X of cell 302 is accessible to the horizontal block connectors 405 (BC-H31) and 408 (BC-H42) through switches 418, 424 respectively. Input A of cell 303 is accessible to the horizontal block connectors 404 (BC-H22) and 406 (BC-H32) through switches 419, 426 respectively, input B of cell 303 is accessible to the horizontal block connectors 403 (BC-H21) and 405 (BC-H31) through switches 420, 425 respectively, output X of cell 303 is accessible to the horizontal block connectors 402 (BC-H12) and 407 (BC-H41) through switches 410 427 respectively. The input A of cell 304 is accessible to the horizontal block connectors 404 (BC-H22) and 406 (BC-H32) through switches 422,430 respectively, input B of cell 304 is accessible to the horizontal block connectors 403 (BC-H21) and 405 (BC-H31) through switches 423, 429

respectively, output X of cell 304 is accessible to the horizontal block connectors 402 (BC-H12) and 407 (BC-H41) through switches 412, 434 respectively. D flip-flop cell 305 input is accessible to the horizontal block connectors 403 (BC-H21) and 406 (BC-H32) through switches 436, 431 respectively, output X of cell 305 is accessible to the horizontal block connectors 401 (BC-H11) and 408 (BC-H42) through switches 432, 435 respectively.

Figures 4A and 4B illustrate the vertical and horizontal block connectors accessing method to the upper left (NW) logical cluster inside a logical block in the currently preferred embodiment. The lower left (SW) cluster has the identical accessing method to the vertical block connectors as those of the NW cluster. The upper right (NE) cluster has similar accessing method to those of the NW cluster with respect to the vertical block connectors except the sequence of vertical block connector access is shifted. The vertical block connectors 451-458 can be viewed as chained together as a cylinder (451, 452, ..., 458). Any shift, say by 4, forms a new sequence: (455, 456, 457, 458, 451, 452, 453, 454). Instead of starting with vertical block connectors 451 and 453 accessing by cell 301 in the NW cluster as illustrated in Figures 4A, the cell 301 in the NE cluster is accessible to VBCs 455 and 457. The numbering is "shifted" by four. The access labeling of the

lower right (SE) cluster to the VBCs is identical to those of NE cluster.

Similarly, the horizontal block connectors access to the NW cluster is identical to those of the NE cluster and the SW cluster is identical to the SE cluster while the horizontal block connectors access to the SW cluster is shifted by four compared with those of NW cluster.

In the currently preferred embodiment, sixteen block connectors are used per logical block (i.e. four clusters, or a 4x4 cell array). Adding a level 1 MLA routing network allows for the connectability for a block cluster (an 8x8 cell array). Adding level 2 MLA routing network increases the connectability to a block sector (16x16 cell array). Additional levels of MLA routing network increases the number of block sectors by factors of four while the length (or reach) of each line in the MLA routing network increases by factors of two. The number of routing lines in the level 2 MLA is increased by a factor of two; since the number of block sectors increased by a factor of four, on a per unit area basis, the number of routing lines in the next level of hierarchy actually decreases by a factor of two.

Figure 5A shows a logical block with associated sixteen block connectors and level 1 MLA routing lines associated with the logical block. The sixteen block connectors 501-516 are depicted by heavy lines whereas the sixteen level 1 MLA

routing network lines 517-532 are depicted by lighter lines. Note that the length or span of the block connectors terminates within the logical block while the length of the level 1 MLA routing network lines extends to neighboring logical blocks (twice the length of the block connectors).

Both block connectors and level 1 MLA routing network lines are subdivided into horizontal and vertical groups: vertical block connectors 501-508, horizontal block connectors 509-516, vertical level 1 MLA routing network lines 517-524, and horizontal level 1 MLA routing network lines 525-532.

In the currently preferred embodiment, there are twenty four level 1 MLA turn points for the sixteen level 1 MLA routing network lines within the logical block. In Figure 5A, the twenty four turn points are depicted as clear dots 541-564.

A MLA turn point is a programmable bi-directional passgate for providing connectability between a horizontal MLA routing network line and a vertical MLA routing network line. For example, enabling level 1 MLA turn point 541 causes the horizontal level 1 MLA routing network line 526 and vertical level 1 MLA routing network line 520 to become connected together. Figure 5B shows level 1 MLA turn point 541. Switch 583 controls whether level 1 MLA routing network line 526 is to be connected to level 1 MLA routing network line 520. If switch is enabled, then level 1 MLA routing network line 526

is connected to level 1 MLA routing network line 520. Otherwise, line 526 is not connected to line 520. Switch 583 is programmable by the user. The turn points are placed as pair-wise groups with the objective of providing switching access connecting two or more block connectors first through the block connector to level 1 MLA exchange networks and then connecting selected level 1 MLA routing lines by enabling the switches. The level 1 MLA lines are used to connect those block connectors that reside in separate logical blocks within the same block cluster.

Referring back to Figure 5A, there are eight block connector to level 1 MLA exchange networks 533-540 for each logical block. These exchange networks operate to connect certain block connectors to level 1 MLA lines as programmed by the user. Figure 5C shows the exchange network 537 in greater detail. The block connector to level 1 MLA routing exchange network has eight drivers 575-582. These eight drivers 575-582 are used to provide bi-directional drive for the block connectors 501, 502 and level 1 MLA lines 517, 518. For example, enabling switch 565 causes the signal on block connector 501 to be driven by driver 575 from the level 1 MLA line 517. Enabling switch 566 causes the signal on level 1 MLA line 517 to be driven by driver 576 from the block connector 501. Enabling switch 567 causes the signal on block connector 501 to be driven by driver 577 from the level

1 MLA line 518. Enabling switch 568 causes the signal on level 1 MLA line 518 to be driven by driver 578 from the block connector 501.

Similarly, enabling switch 569 causes the signal on block connector 502 to be driven by driver 579 from the level 1 MLA line 517. Enabling switch 570 causes the signal on level 1 MLA line 517 to be driven by driver 580 from the block connector 502. Enabling switch 571 causes the signal on block connector 502 to be driven by driver 581 from the level 1 MLA line 518. Enabling switch 572 causes the signal on level 1 MLA line 518 to be driven by driver 582 from the block connector 502. Switch 573 is used to control whether a signal should pass form one block connector 501 to the adjacent block connector 584 belonging to the adjacent logical block.

Likewise, switch 574 is used to control whether a signal should pass form one block connector 502 to the adjacent block connector 585 belonging to the adjacent logical block.

Figure 6 shows the routing network for a block cluster. The block cluster is basically comprised of four logical blocks which can be interconnected by the level 1 MLA exchange networks 533-540. It can be seen that there are thirty-two level 1 MLA routing network lines.

Figure 7A shows the block diagram for a block sector. The block sector is comprised of four block clusters 701-704.

As discussed above, the block clusters are interconnected by block connectors and level 1 MLA routing network lines. In addition, the block sector is also comprised of sixty-four level 2 MLA routing network lines and sixty-four level 2 to level 1 exchange networks to provide connectability between level 1 MLA routing network and level 2 MLA routing network. The level 1 to level 2 MLA routing exchange networks are depicted by rectangles in Figure 7A. Furthermore, there are forty-eight level 2 MLA turn points associated with each of the four logical blocks within the block sector. Consequently, there are one hundred and ninety-two level 2 MLA turn points for the block sector.

Figure 7B shows a sample level 1 to level 2 MLA routing exchange network 705. It can be seen that switch 710 is used to control whether a signal should pass between level 1 MLA line 709 and level 2 MLA line 708. Switch 711 is used to control whether a signal should pass between level 1 MLA line 709 and level 2 MLA line 707. Switch 712 is used to control whether a signal should pass between level 1 MLA line 706 and level 2 MLA line 708. Switch 713 is used to control whether a signal should pass between level 1 MLA line 706 and level 2 MLA line 707. Switch 714 is used to control whether a signal should pass form one level 1 MLA line 709 to the adjacent level 1 MLA line 716 belonging to the adjacent block cluster. Likewise, switch 715 is used to control whether

a signal should pass form one level 1 MLA line 706 to the adjacent level 1 MLA line 715 belonging to the adjacent block cluster.

Figure 8A shows a sector cluster. The sector cluster is comprised of four block sectors 801-804 with their associated block connectors, level 1, and level 2 MLA routing network lines and exchange networks. In addition, there are one hundred and twenty-eight level 3 MLA routing network lines, providing connectability between the level 2 MLA lines that belong to different block sectors 801-804 within the same sector cluster 800. There are ninety-six level 3 MLA turn points associated with the level 3 MLA lines for each of the block sector 801-804 (i.e. three hundred and eighty-four total level 3 MLA turn points for the sector cluster). Furthermore, there are thirty-two level 2 to level 3 MLA routing exchange networks associated with each of the four block sector 801-804. Hence, there are total of one hundred and twenty-eight level 3 MLA routing exchange network for providing programmable connectability between the various level 2 and level 3 MLA lines.

Figure 8B shows an example of a level 2 to level 3 MLA routing exchange network 805. It can be seen that enabling switch 810 causes a signal on the level 2 MLA line 808 to be connected to the level 3 MLA line 806. Disabling switch 810 disconnects the level 2 MLA line 808 from the level 3 MLA

line 806. Enabling switch 811 causes a signal on the level 2 MLA line 808 to be connected to the level 3 MLA line 807. Disabling switch 811 disconnects the level 2 MLA line 808 from the level 3 MLA line 807. Likewise, enabling switch 812 causes a signal on the level 2 MLA line 809 to be connected to the level 3 MLA line 806. Disabling switch 812 disconnects the level 2 MLA line 809 from the level 3 MLA line 806. Enabling switch 813 causes a signal on the level 2 MLA line 809 to be connected to the level 3 MLA line 807. Disabling switch 813 disconnects the level 2 MLA line 809 from the level 3 MLA line 807.

In the present invention, larger and more powerful FPGAs can be achieved by adding additional logic sector clusters which are connected by additional levels of MLA routing networks with the corresponding MLA turn points and exchange networks.

In one embodiment of the present invention, each of the five I-Matrix lines (331-335, Figure 3A) can be extended to provide connectability between two adjacent I-Matrix lines belonging to two different clusters. The passgate switches 336-340, 341-345, 346-350, and 351-355 in Figure 3B are examples of four different sets of I-Matrix line extension switches. This provides further flexibility by providing the capability of routing a signal between two adjacent clusters

without having to be routed through the use of block connectors.

Similarly, block connectors can be extended to provide connectability between two adjacent block connectors belonging to two different logical blocks. Switch 573 of Figure 5C illustrates such block connector extension connecting block connector 501 to block connector 584 through switch 573. This provides further flexibility by providing the capability of routing a signal between two adjacent logical blocks without having to be routed through the level 1 MLA lines and associated MLA exchange networks. This concept can be similarly applied to the level 1 MLA lines as well. Switch 714 of Figure 7B shows an example where level 1 MLA line 709 is extended to connect to level 1 MLA line 716 by enabling switch 714. This provides further flexibility by providing the capability of routing a signal between two adjacent block clusters without having to be routed through the level 2 MLA lines and associated MLA exchange networks.

Figure 9 shows one embodiment of a hierarchical multiple level routing network for providing routability between the logical blocks and the MLA levels. Eight logical blocks 901-908 are shown. Associated with each of the logical blocks 901-908 are a plurality of block connectors. In the currently preferred embodiment, there are eight horizontal and eight vertical block connectors associated with each of the

logical blocks 901-908. For clarity and ease of comprehension, the block connectors corresponding to an individual logical block is represented by a single line (e.g., block connectors 909-916 respectively correspond to logical blocks 901-908) and only the horizontal block connectors are shown.

In turn, each of the block connectors 909-916 are respectively coupled to programmable bi-directional drivers 917-924. Consequently, block connectors 909-916 can be programmed to be coupled bi-directionally to the MLA-1 lines 925-928. For example, exchange network 917 can be programmed to couple one of the block connectors 909 of logical block 901 to the MLA-1 line 925. Additional programmable bi-directional drivers 929-932 are used to provide interconnections between the MLA-1 lines 925-928 and the next MLA level, MLA-2 lines 933-934. Programmable bi-directional drivers 935-936 selectively provide interconnections between the MLA-2 lines 933-934 and the MLA-3 line 937. This hierarchical interconnection scheme can be repeated for additional multiple MLA levels.

Figure 10 shows another embodiment of a hierarchical multiple level routing network for providing routability between the logical blocks and the MLA levels. This embodiment is similar to the routing network shown in Figure 9, except that the block connectors can be directly connected to any of the MLA levels and bypassing any intervening MLA

level. Eight logical blocks 1001-1008 are shown. Associated with each logical block are a plurality of block connectors 1009-1016. Programmable bi-directional drivers 1017-1024 are used to selectively couple the block connectors 1009-1016 to the block connector tabs 1025-1032. The block connector tabs 1025-1032 are used as junction points from which connections can be made to multiple MLA layers. Programmable bi-directional driver sets (1033-1035), (1036-1038), (1039-1041), (1042-1044), (1045-1047), (1048-1050), (1051-1053), (1054-1056), correspond to block connector tabs 1025-1032, respectively. Each of these driver sets enables their respective logical block to be connected to either the MLA-1 line 1061, MLA-2 line 1062, or MLA-3 line 1063 without requiring it to pass through any intervening MLA lines. For example, logical block 1001 can be connected to the MLA-1 line 1061 by selectively activating drivers 1017 and 1033. Logical block 1001 can also be connected to the MLA-2 line 1062 by selectively activating drivers 1017 and 1034. Note that in this embodiment, logical block 1001 can be connected to the MLA-2 line 1062 without having to first be connected to the MLA-1 line 1061. Furthermore, logical block 1001 can be connected to the MLA-3 line 1063 by selectively activating drivers 1017 and 1035. Note that in this embodiment, logical block 1001 need not be connected to either the MLA-1 nor the MLA-2 layers in order for it to be

connected to the MLA-3 layer. By directly connecting the logical block to the desired MLA layer, the speed of the overall routing network is improved. Furthermore, speed and routing flexibility can be enhanced by directly connecting two or more adjacent logical blocks. Thereby, adjacent logical blocks can communicate without having to be routed over any of the MLA layers. For example, logical blocks 1001 and 1002 can be connected together via the programmable bi-directional driver 1057; logical blocks 1003 and 1004 can be connected via driver 1058; and logical blocks 1005-1007 can be connected via drivers 1059-1060. This hierarchical routing scheme can readily be any number of logical blocks and MLA layers. In addition, a passgate 1064 can be included to couple block connector 1010 corresponding to logic block 1002 to block connector 1011 corresponding to logic block 1003.

Figure 11 shows a block diagram of one embodiment of the hierarchical routing network wherein two groups of block connectors access the same MLA lines. A first group of logical blocks 1101-1104 and a second group of logical blocks 1105-1108 are shown. The first group of logical blocks 1101-1104 can be selectively connected to the MLA-1 layer 1109 and 1121, MLA-2 layer 1110, MLA-3 layer 1111, and MLA Tab 1112 via block connector tabs 1113-1116. Similarly, the second group of logical blocks 1105-1108 can be selectively connected to the MLA-1 layer 1109 and 1121, MLA-2 layer

1110, MLA-3 layer 1111, and MLA Tab 1112 via their respective block connector tabs 1117-1120.

Figure 12 shows a block diagram of part of the multiple level routing network which encompasses Block Connectors to the MLA-3 Level with MLA Tabs for higher levels of routing network (the I-Matrix is not shown). Figure 12 shows the interconnections of one set of Block Connectors and its corresponding higher levels of MLAs in the horizontal direction. There is also a corresponding perpendicular (e.g., vertical) group of routing network interconnecting the Block Connectors and the associated MLAs. This perpendicular group is not shown in Figure 12 in order to avoid obscuring the present invention. Note that there is a corresponding copy of the routing network for each and every Block Connector and associated MLAs of the FPGA.

Shown in Figure 12 are thirty-two blocks 1201-1232. Each block is associated with a distinct and adjacent block along with two BC tabs (e.g., one horizontal and one vertical). Each of the Block Connectors 1201-1232 are coupled to two selectable BC Tabs via a programmable switch. For example, block connector 1201 is coupled to selectable BC Tab 1233 through programmable switch 1234. The second group of BC Tabs, which is perpendicular (e.g., vertical) to the first BC Tab group is not shown. A similar BC Tab interconnection scheme exists for block connectors 1217-1232 (both horizontally and

vertically). For each BC Tab, there are bi-directional programmable drivers connectable to the MLA-1 routing lines. For example, BC Tab 1233 is selectively connectable to the MLA-1 routing line 1235 via drivers 1236. These drivers can either be parallel to or perpendicular to the corresponding BC Tabs. In the currently preferred embodiment, the number of MLA-1 lines is half the number of Block Connectors, since for each Block Connector, there is a corresponding MLA-1 line plus another MLA-1 line which is perpendicular to the first MLA-1 line. Each MLA-1 line is connectable through programmable means to the corresponding Block Connector, MLA-2, and MLA-3 lines through their corresponding BC Tab. Note that the MLA-1 routing network together with I-Matrix lines and Block Connectors form the routing resources in a 2 x 2 Block area. This format enhances more complex logic function formation accessing and interconnecting the cells. Furthermore, the MLA-1 routing network, in addition to both I-Matrix lines and Block Connectors, become additional bi-directionally programmable access lines that can serve as access ports for the implementation of even more complex logic functions through connections by other MLA lines or Block Connectors from outside of the 2 x 2 Block area. By using programmable switches, the I-Matrix lines and block connectors can be selectively accessed which are not necessarily adjacent or congruent to the 2 x 2 Block areas.

Hence, the total number of routing segments including I-Matrix lines, Block connectors, and MLA-1 lines grow geometrically when the growth is from a Block to 2 x 2 Blocks.

For each BC Tab, there is bi-directionally programmable drivers connectable to the MLA-2 routing lines. For example, block connector tab 1233 is connectable to the MLA-2 line 1237 via drivers 1238. The MLA-2 can either be parallel to or perpendicular to the corresponding BC Tabs. In the currently preferred embodiment, the number of MLA-2 lines is half the number of MLA-1 lines. Each MLA-2 line is connectable through programmable means to the corresponding Block Connector, MLA-1, and MLA-3 lines through the corresponding BC Tab. The MLA-2 routing network together with I-Matrix lines, Block Connectors and MLA-1 routing network form the routing resources in a 4 x 4 Block area for more complex logic function formation accessing and for interconnecting the cells. In this case, the MLA-2 routing network, in conjunction with the I-Matrix lines, Block Connectors and MLA-1 lines, become additional bi-directionally programmable access lines that can serve as access ports for the implementation of even more complex logic functions through connections with other MLA lines or Block connectors from outside of the 4 x 4 Block area. By means of programmable switches, the access need not necessarily be adjacent or congruent to the 4 x 4 Block area.

The total number of routing segments including I-Matrix lines, Block Connectors, MLA-1 lines, and MLA-2 lines in a 4 x 4 Block unit grows proportional to the increase in logic cells. The increase in the total number is geometrical when the growth is from a Block to 4 x 4 Blocks. Similarly, for each BC Tab, there is bi-directionally programmable drivers connectable to MLA-3 routing lines. For example, BC Tab 1233 is connectable to the MLA-3 line 1239 via drivers 1240. The MLA-3 routing line can either be parallel to or perpendicular (e.g., horizontal or vertical) to the corresponding BC Tabs. In the currently preferred embodiment, the number of MLA-3 lines is half the number of MLA-2 lines. Each MLA-3 line is connectable through programmable means to the corresponding Block Connector, MLA-1, and MLA-2 lines through the corresponding BC Tab. The MLA-3 routing network together with I-Matrix lines, Block Connectors, MLA-1 routing network and MLA-2 routing network form the routing resources in a 8 x 8 Block area for more complex logic function formation accessing and interconnecting the cells. The MLA-3 routing network, in addition to both I-Matrix lines, Block Connectors, MLA-1 lines and MLA-2 lines, become additional bi-directionally programmable access lines that can serve as access ports for the implementation of even more complex logic functions through connections by other MLA lines or Block Connectors that are outside of the 8 x 8 Block

area and are not necessarily adjacent or congruent to the 8 x 8 Block area through programmable means. Hence, the total number of routing segment including I-Matrix lines, Block Connectors, MLA-1 lines, MLA-2 lines and MLA-3 lines in an 8 x 8 Block unit grows proportional to the increase in logic cells. This increase is geometrical when the growth is from a Block to 8 x 8 Blocks. In addition, for each BC Tab, there is bi-directionally programmable drivers connectable to MLA Tabs. For example, BC Tab 1233 is connectable to the MLA Tab 1241 via drivers 1242. The MLA Tabs can either be parallel to or perpendicular to the corresponding BC Tabs. Each bi-directionally programmable driver (e.g., driver's 1236, 1238, 1240, 1242, etc.) can be either passgate controlled through programmable means; bi-directional drivers with passgates controlled through programmable means; a tri-state controlled through programmable means in one direction and passgate or driver with a passgate controlled through programmable means; or two tri-states in opposite directions controlled through programmable means. The choice is a function of speed and density requirements.

In one embodiment, each Block Connector and BC Tab have extensions to the adjacent Blocks. For example, block 1201 is connectable to block 1202 via programmable switch 1243. BC Tab 1244 is connectable to BC Tab 1245 via programmable switch 1246. It should be noted that additional

extensions for MLA lines can be implemented in order to extend the routing range without having to use higher level MLA lines. Multiple variations to the routing network shown in Figure 12 are possible. For example, to increase routing resources and hence routability, the MLA-1 routing network can be replaced by making two copies of the MLA-2 routing network. On the other hand, if the objective is to minimize the routing area, one embodiment minimizes the amount of programming bits by replacing the MLA-1 routing network with a copy of the MLA-2 routing network. These kinds of variations can be applied to a mixture of other levels. Another embodiment is to off-set one or more of the MLA lines. For example, in Figure 12, the 1247 is accessible by BC Tabs 1245, 1248, 1249, and 1250. The MLA-1 line 1247 can be shifted by one block to become accessible by BC Tabs 1248, 1251, 1250, and 1252 instead. All other MLA-1 lines can be thusly shifted. This can also be applied to other MLA level(s).

Figure 13 shows an MLA-1 turn network. Four logical blocks 1301-1304 are shown. These four logical blocks are connected to each of the MLA-1 lines of sets 1305-1308. Each of the MLA-1 lines is connectable through a programmable means (e.g., turn points 1309) to all the perpendicular MLA-1 lines, except the corresponding perpendicular MLA-1 line. For example, the horizontal MLA-1 line 1310 is connectable to the vertical MLA-1 line 1311 via turn point 1312. The purpose

for the MLA-1 lines is to connect a set of Block Connectors together that is within the MLA-1 routing network range. In the case of connecting corresponding Block Connectors within a four-Blocks area, as shown in Figure 13, the connection(s) can be made through either the Block Connector extension or through a BC Tab to one of the corresponding MLA-1 line, without having to resort to using two perpendicular MLA-2 lines through turn points. In one embodiment, the number of turn points is reduced. This restricts the turn flexibility but also reduces both the loading on the MLA-1 lines and the area required to lay out the design. However, routing flexibility and routability may be affected.

Figure 14 shows an MLA-2 turn network. As can be seen, each MLA-2 line is connectable through programmable means to every MLA-2 lines which are perpendicular to the MLA-2 line. For example, the vertical MLA-2 line 1401 is connectable to the horizontal MLA-2 line 1402 through turn point 1403. In other embodiments, the turn flexibility can be made more restrictive by reducing the number of turn points. This will reduce both the loading on the MLA-2 line and the area required to lay out the design. However, routing flexibility and routability may be affected.

Figure 15 shows an MLA-3 turn network. Each MLA-3 line is connectable through programmable means to all the perpendicular MLA-3 lines. For example, the vertical MLA-3

line 1501 is connectable to the horizontal MLA-3 line 1502 through turn point 1503. The turn flexibility can be made more restrictive by reducing the number of turn points. This will reduce both the loading on the MLA-3 line and the area required to lay out the design. However, routing flexibility and routability may be affected.

Figure 16 shows one embodiment of a routing network for the MLA-4 layer and the mechanism whereby the MLA-4 lines are accessed. Figure 16 shows four 8 x 8 Blocks 1621-1624 (for a total of 16 x 16 Blocks). Associated with the four 8 x 8 Blocks 1621-1624 are four horizontal and four vertical groups of MLA Tabs. In the currently preferred embodiment, the MLA-4 lines and MLA Tabs are 8-bits wide. Since each Block has eight corresponding Block Connectors, each MLA Tab is shown to be eight lines wide where each of the lines corresponds to one of the 8 Block Connectors as shown earlier in Figure 12. In the currently preferred embodiment, there are four vertical and four horizontal MLA-4 lines, each of which is eight lines wide. Thus, the number of MLA-4 lines is one-fourth the number of MLA-3 lines. Each MLA-4 line is connectable through programmable means to the corresponding Block Connector, MLA-1, MLA-2 and MLA-3 lines. The desired connectivity is made through the corresponding MLA Tab and the BC Tab. The MLA-4 routing network together with the I-Matrix lines, Block Connectors,

MLA-1 routing network, MLA-2 routing network and MLA-3 routing network, form the routing resources in a 16 x 16 Block area for more complex logic function formation accessing and interconnecting of the cells. In one embodiment, the MLA-4 routing network, in addition to both I-Matrix lines, Block Connectors, MLA-1 lines, MLA-2 lines and MLA-3 lines become additional bi-directionally programmable access lines that can serve as access ports for the implementation of even more complex logic functions through connections by other MLA lines of Block Connectors from outside of the 16 x 16 Block area through programmable means. These other MLA lines or block connectors need not necessarily be adjacent or congruent to the 16 x 16 Block area. The total number of routing segments including I-Matrix lines, Block connectors, MLA-1 lines, MLA-2 lines, MLA-3 lines and MLA-4 lines in a 16 x 16 Block unit grows proportional to the increase in logic cells. The increase in size is geometrical when the growth is from a Block to 16 x 16 Blocks. From each MLA Tab there is a corresponding MLA-4 line connectable to the MLA Tab via a switch. For example, MLA Tab 1601 is connectable to MLA-4 line 1602 via switch 1603. Similarly, MLA Tab 1601 is connectable to MLA-4 line 1604 via switch 1605; MLA-4 line 1606 via switch 1607; and MLA-4 line 1608 via switch 1609. Likewise, MLA Tab 1610 is connectable to MLA-4 lines 1611-1614 via switches 1615-1618, respectively. Each MLA Tab in

any one of the four corners is connectable through programmable means to all the corresponding MLA Tabs in all four corners through the vertical or the horizontal MLA-4 lines.

Figure 17 shows three different switch embodiments 1701-1703. In general, the switch is a bi-directionally programmable driver network which can be a simple bi-directional passgate, or any of the bi-directional driver configurations 1701-1703.

Figure 18 shows one embodiment of a routing network for the MLA-5 layer and the mechanism whereby MLA-5 lines are accessed. Sixteen 8 x 8 Blocks are shown. Associated with each of the 8 x 8 Blocks are four horizontal and four vertical MLA Tabs, which are the same as the MLA Tabs shown in Figure 16. When the 16 x 16 Blocks (as shown in Figure 16) are grouped as a unit, the next higher level, which consists of 32 x 32 Blocks, is formed. Associated with each of the four 16 x 16 corner units are four horizontal and four vertical MLA-5 lines, each 8-bit wide. These lines are shared by the adjacent corner units, as shown in Figure 18. Thus, the number of MLA-5 lines is half the number of MLA-4 lines. Each MLA-5 line is connectable through programmable means to the corresponding Block Connector, MLA-1, MLA-2, MLA-3 and MLA-4 lines through the corresponding MLA Tab and the BC Tab. The MLA-5 routing network together with I-Matrix lines,

Block Connectors, MLA-1 routing network, MLA-2 routing network, MLA-3 routing network and MLA-4 routing network form the routing resources in a 32 x 32 Block area for more complex logic function formation accessing and interconnecting the cells. Furthermore, the MLA-5 routing network, in addition to both I-Matrix lines, Block Connectors, MLA-1 lines, MLA-2 lines, MLA-3 lines and MLA-4 lines can be used as additional bi-directionally programmable access lines that can serve as access ports for the implementation of even more complex logic functions through connections by other MLA lines or Block Connectors from outside of the 32 x 32 Block area (which need not necessarily be adjacent or congruent to the 32 x 32 Block area) through programmable means. The total number of routing segments including I-Matrix lines, Block Connectors, MLA-1 lines, MLA-2 lines, MLA-3 lines, MLA-4 lines and MLA-5 lines in a 32 x 32 Block unit grows proportional to the increase in logic cells. This increase is geometrical when the growth is from a Block to 32 x 32 Blocks.

From each MLA Tab there is a corresponding MLA-5 line connectable to the MLA Tab via a switch. The switch is a bi-directionally programmable driver network which can be a simple bi-directional passgate, or any of the bi-directional driver configurations as shown in Figure 17. In addition, turn points are incorporated where the vertical MLA-5 lines intersect the horizontal MLA-5 lines through programmable

means. Each MLA Tab in any one of the four corners is connectable through programmable means to all the corresponding MLA Tabs in all four corners. This is implemented by a combination of programmable connections to the vertical and the horizontal MLA-5 lines plus use of the turn points.

Higher levels of MLA networks can be developed by programmable access through the MLA Tabs or by introducing another new intermediate MLA Tabs. In such instances, the number of MLA lines is a fraction of the next lower level MLAs. The total number of routing segments including I-Matrix lines, Block Connectors, MLA-1 lines, MLA-2 lines, MLA-3 lines, MLA-4 lines, MLA-5 lines and higher levels of MLA lines, and the corresponding number of n x n Block unit grows proportional to the increase in logic cells. This increase is geometrical when the growth is from a Block to the n x n Blocks.

Thus, an architecture with an intraconnect and interconnect scheme for programmable logic circuits is disclosed.

CLAIMS

What is claimed is:

1. A programmable logic circuit comprising:

a plurality of logic blocks having a plurality of programmable interconnected cells for performing logic functions on logic signals;

a first set of programmable switches coupled to said plurality of cells of said plurality of logic blocks;

a first set of routing lines coupled to said first set of programmable switches to form a first set of bi-directionally programmable access lines to said plurality of logic blocks, wherein said first set of bi-directionally programmable access lines function as input/output pins for said plurality of logic blocks through programmable means;

a second set of programmable switches coupled to said first set of bi-directionally programmable access lines;

a second set of routing lines coupled to said second set of programmable switches.

2. The programmable logic circuit of Claim 1 further comprising:

a third set of programmable switches for selectively coupling a plurality of lines of said second set of routing lines, wherein said second set of routing lines and said third set of

programmable switches comprise a first level of interconnections;

a third set of routing lines connectable to said second set of programmable switches.

3. The programmable logic circuit of Claim 2 further comprising:

a fourth set of programmable switches for selectively coupling a plurality of lines of said third set of routing lines, wherein said third set of routing lines and said fourth set of programmable switches comprise a second level of interconnections;

a fifth set of programmable switches connectable between said third set of routing lines and said second set of routing lines for selectively coupling said plurality of logic blocks to said second level of interconnections via said first level of interconnections.

4. The programmable logic circuit of Claim 1 further comprising:

a sixth set of programmable switches for selectively coupling a line of said first set of bi-directionally programmable access lines of a first logic block to a line of said first set of bi-directionally programmable access lines of a

second logic block, wherein first said logic block is adjacent to said second logic block.

5. The programmable logic circuit of Claim 4 wherein the number of lines of said first set of routing lines for each logic block of said plurality of logic blocks is approximately half the number of said plurality of cells of said logic block.

6. The programmable logic circuit of Claim 2, wherein the span of each of the said second set of routing lines is twice the span of each of said first set of routing lines.

7. The programmable logic circuit of Claim 2, wherein the number of said second set of routing lines is half the number of said first set of routing lines.

8. The programmable logic circuit of Claim 3, wherein the span of each of said third set of routing lines is twice the span of each of said second set of routing lines.

9. The programmable logic circuit of Claim 3, wherein the number of said third set of routing lines is half the number of the said second set of routing lines.

10. The programmable logic circuit of Claim 1 further comprising:

a fourth set of routing lines connectable to said second set of programmable switches;

a seventh set of programmable switches for selectively coupling a plurality of lines of said fourth set of routing lines, wherein said fourth set of routing lines and said seventh set of programmable switches comprise a third level of interconnections;

an eighth set of programmable switches connectable between said fourth set of routing lines and said third set of routing lines for selectively coupling said plurality of logic blocks to said third level of interconnections via said second level of interconnections or said first level of interconnections.

11. The programmable logic circuit of Claim 10, wherein the span of each of said fourth set of routing lines is twice the span of each of said third set of routing lines and the number of said fourth set of routing lines is half the number of said third set of routing lines.

12. The programmable logic circuit of Claim 10, wherein said fourth set of routing lines and said seventh set of programmable switches comprise a third level of interconnections.

13. The programmable logic circuit of Claim 1 further comprising:

a fifth set of routing lines, connectable to said second set of programmable switches for forming a fourth level of interconnections.

14. The programmable logic circuit of Claim 13, wherein the span of each of said fifth set of routing lines is twice the span of each of said third set of routing lines.

15. The programmable logic circuit of Claim 13, wherein the number of said fifth set of routing lines is half of the number of said third set of routing lines.

16. The programmable logic circuit of Claim 1 further comprising:

a sixth set of routing lines comprising a fifth level of interconnections;

a ninth set of programmable switches connectable between said sixth set of routing lines and said fifth set of routing lines capable of conducting signal from said first set of bi-directionally programmable access lines for selectively coupling said plurality of logic blocks to said fifth level of interconnections via said fourth level of interconnections and

bypassing said first level of interconnections, said second level of interconnections, and said third level of interconnections.

17. The programmable logic circuit of Claim 16, wherein the span of each of said sixth set of routing lines is twice the span of each of said fourth set of routing lines.

18. The programmable logic circuit of Claim 16, wherein the number of said sixth set of routing lines is one fourth the number of said fourth set of routing lines.

19. The programmable logic circuit of Claim 1 further comprising:

a seventh set of routing lines;

a tenth set of programmable switches coupled for selectively connecting a plurality of lines of said seventh set of routing lines, wherein said seventh set of routing lines and said tenth set of programmable switches comprise a sixth level of interconnections;

an eleventh set of programmable switches connectable between said seventh set of routing lines and said fifth set of routing lines capable of conducting signal from said first set of bi-directionally programmable access lines for selectively coupling said plurality of logic via said fourth level of interconnections and bypassing said first level of

interconnections, said second level of interconnections, said third level of interconnections, and said fifth level of interconnections.

20. The programmable logic circuit of Claim 19, wherein the span of each of said seventh set of routing lines is twice the span of each of said sixth set of routing lines.

21. The programmable logic circuit of Claim 19, wherein the number of said seventh set of routing lines is half the number of said sixth set of routing lines.

22. The programmable logic circuit of Claim 1, wherein said first level of interconnections function as a second set of bi-directionally programmable access lines.

23. The programmable logic circuit of Claim 2, wherein said second level of interconnections function as third set of bi-directionally programmable access lines.

24. The programmable logic circuit of Claim 1, wherein said switches are comprised of bi-directionally programmable drivers.

25. The programmable logic circuit of Claim 1, wherein said switches are comprised of bi-directionally programmable passgates.

26. The programmable logic circuit of Claim 1 further comprising a twelfth set of programmable switches for selectively coupling an adjacent set of first set of bi-directionally programmable access lines of an adjacent set of logic blocks to said same first level of interconnections.

27. The programmable logic circuit of Claim 1 further comprising a thirteenth set of programmable switches for selectively coupling an adjacent set of first set of bi-directionally programmable access lines of an adjacent set of logic blocks to a same second level of interconnections.

28. The programmable logic circuit of Claim 10 further comprising a fourteenth set of programmable switches for selectively coupling an adjacent set of first set of bi-directionally programmable access lines of an adjacent set of logic blocks to said same third level of interconnections.

29. The programmable logic circuit of Claim 13 further comprising a fifteenth set of programmable switches for selectively coupling an adjacent set of first set of bi-

directionally programmable access lines of an adjacent set of logic blocks to a same fourth level of interconnections.

30. The programmable logic circuit of Claim 16 further comprising a sixteenth set of programmable switches for selectively coupling an adjacent set of said fourth level of interconnections to a same fifth level of interconnections.

31. The programmable logic circuit of Claim 19 further comprising a seventeenth set of programmable switches for selectively coupling an adjacent set of said fourth level of interconnections to a same sixth level of interconnections.

32. The programmable logic circuit of Claim 10, wherein said third level of interconnections function as fourth set of bi-directionally programmable access lines.

33. The programmable logic circuit of Claim 13, wherein said fourth level of interconnections function as fifth set of bi-directionally programmable access lines.

34. The programmable logic circuit of Claim 16, wherein said fifth level of interconnections function as sixth set of bi-directionally programmable access lines.

35. The programmable logic circuit of Claim 19, wherein said sixth level of interconnections function as seventh set of bi-directionally programmable access lines.

36. A programmable logic circuit of Claim 1 further comprising:

a plurality of logic clusters, each of said logic clusters having a plurality of programmable interconnected cells for performing logic functions on logic signals;

an eighteenth set of programmable switches coupled to said plurality of logic cells of a said logic cluster;

an eighth set of routing lines coupled to said eighteenth set of programmable switches to form a set of intraconnection matrix routing lines coupled to said logic cluster.

37. The programmable logic circuit of Claim 36, wherein said set of intraconnection matrix routing lines function as short distance connections for the said plurality of logic cells of the said logic cluster through programmable means.

38. The programmable logic circuit of Claim 36, the number of lines in each said set of intraconnection matrix routing lines is equal to approximately half the number of

total number of input pins and output pins of the said plurality of cells in said logic cluster.

39. The programmable logic circuit of Claim 36, the span for each line of said set of intraconnection matrix routing lines is equal to half the span of each line of said first set of bi-directionally programmable access lines.

40. The programmable logic circuit of Claim 36, said cells of a said logic block are comprised of said plurality of cells of said plurality of logic clusters.

41. The programmable logic circuit of Claim 36 further comprising a nineteenth set of programmable switches for selectively coupling a line of said set of intraconnection matrix of a first logic cluster to a line of the said set of intraconnection matrix of a second logic cluster, wherein said first logic cluster is adjacent to said second logic cluster.

42. A programmable logic circuit comprising:

a plurality of logic clusters, each of said logic clusters having a plurality of programmably interconnected cells for performing logic functions on logic signals;

a set of intraconnection matrix routing lines coupling a set of said plurality of logic clusters;

a plurality of logic blocks, each of said logic blocks having a plurality of programmably interconnected cells for performing logic functions on logic signals;

a first set of bi-directionally programmable access lines coupling a first set of said plurality of logic blocks.

43. The programmable logic circuit of Claim 42 further comprising a first level of interconnections coupling at least two of said first set of bi-directionally programmable access lines.

44. The programmable logic circuit of Claim 43 further comprising a first programmable switch coupled to a bi-directionally programmable access line of said first set of bi-directionally programmable access lines of a first logic block for programmably conducting a signal corresponding to said first logic block to said first level of interconnections.

45. The programmable logic circuit of Claim 44 further comprising a second level of interconnections coupling at least two of said first level of interconnections.

46. The programmable logic circuit of Claim 45 further comprising a second programmable switch connectable to said bi-directionally programmable access line of said first set of bi-directionally programmable access lines of said first logic block for programmably conducting said signal corresponding to said first logic block to said second level of interconnections, wherein said first level of interconnections is capable of being bypassed by said signal.

47. The programmable logic circuit of Claim 46 further comprising:

a third level of interconnections coupling at least two of said second level of interconnections.

48. The programmable logic circuit of Claim 47 further comprising a third programmable switch connectable to said bi-directionally programmable access line of said first set of bi-directionally programmable access lines of said first logic block for programmably conducting said signal corresponding to said first logic block to said third level of interconnections, wherein said first level of interconnections and said second level of interconnections are capable of being bypassed by said signal.

49. The programmable logic circuit of Claim 42 further comprising:

a fourth level of interconnections;

a routing line coupled to said fourth level of interconnections;

a fourth programmable switch connectable to said bi-directionally programmable access line of said first set of bi-directionally programmable access lines of said first logic block for programmably conducting said signal corresponding to said first logic block to said routing line.

50. The programmable logic circuit of Claim 42 further comprising a fifth programmable switch connectable to said intraconnection matrix routing line of said set of intraconnection matrix routing lines of said first logic cluster for programmably conducting said signal corresponding to said intraconnection matrix routing line of said set of intraconnection matrix routing lines of said first logic cluster to a intraconnection matrix routing line of said set of intraconnection matrix routing lines of a second logic cluster, wherein said first logic cluster is adjacent to said second logic cluster.

51. The programmable logic circuit of Claim 42 further comprising a fifth programmable switch connectable to said

bi-directionally programmable access line of said first set of bi-directionally programmable access lines of said first logic block for programmably conducting said signal corresponding to said bi-directionally programmable access line of said first set of bi-directionally programmable access lines of said first logic block to a bi-directionally programmable access line of a first set of bi-directionally programmable access lines of a second logic block, wherein said first level of interconnections are capable of being bypassed and wherein said first logic block is adjacent to said second logic block.

52. The programmable logic circuit of Claim 42 wherein said switches are comprised of bi-directionally programmable drivers.

53. The programmable logic circuit of Claim 42 wherein said switches are comprised of bi-directionally programmable passgates.

54. The programmable logic circuit of Claim 43 further comprising a sixth set of programmable switches for selectively coupling an adjacent set of first set of bi-directionally programmable access lines of the adjacent set of logic blocks to said first level of interconnections.

55. In a programmable logic circuit having a plurality of cells for performing logic functions on logic signals, a method of forming complex logic functions for programmably coupling a group of said cells by routing signals from one of said cells to another of said cells, said method comprising the steps of:

programmably interconnecting a plurality of said cells to form a first logic cluster, a second logic cluster, and a third logic cluster via a set of intraconnection matrix routing lines;

programmably conducting a signal corresponding to a first set of intraconnection matrix routing lines of a first logic cluster to a first set of intraconnection matrix routing lines of said second logic cluster via a first set of bi-directionally programmable access lines;

programmably interconnecting a plurality of said cells to form a first logic block, a second logic block, and a third logic block;

programmably conducting a signal corresponding to a first set of bi-directionally programmable access lines of a first logic block to a first set of bi-directionally programmable access lines of said second logic block via a first level of interconnections.

56. The method of Claim 55, wherein the span of each of the said first set of intraconnection matrix routing lines is a fraction of the span of each of the said first set of bi-directionally programmable access lines and the number of lines of the said set of intraconnection matrix routing lines is a fraction of the number of input and output pins of said cells in said logic cluster.

57. The method of Claim 55, wherein the span of each of the said first set of bi-directionally programmable access lines is a fraction of the span of each of the said first level of interconnections lines and the number of lines of the said first level of interconnections lines is a fraction of the number of the said first set of bi-directionally programmable access lines over a logic block circuit area.

58. The method of Claim 55 further comprising the step of:

programmably conducting said signal to a first set of bi-directionally programmable access lines of said third logic block via a second level of interconnections by conducting said signal through said first level of interconnections.

59. The method of Claim 55 further comprising the step of:

programmably conducting said signal to a first set of bi-directionally programmable access lines of said third logic block via a second level of interconnections without conducting said signal through said first level of interconnections.

60. The method of Claim 58, wherein the span of each of the said first level of interconnections lines is a fraction of the span of each of said second level of interconnections lines and the number of lines of the said second level of interconnections lines is a fraction of the number of the said first level of interconnections lines over total logic blocks circuit area.

61. The method of Claim 55 further comprising the step of programmably conducting said signal to a first set of bi-directionally programmable access lines of a fourth logic block via a third level of interconnections by conducting said signal through said first level of interconnections or said second level of interconnections.

62. The method of Claim 55 further comprising the step of programmably conducting said signal to a first set of bi-directionally programmable access lines of a fourth logic block via a third level of interconnections without conducting

said signal through said first level of interconnections or said second level of interconnections.

63. The method of Claim 62, wherein the span of each of the said second level of interconnections lines is a fraction of the span of each of the said third level of interconnections lines and the number of lines of the said third level of interconnections lines is a fraction of the number of the said second level of interconnections lines over a logic block circuit area.

64. The method of Claim 55 further comprising the step of programmably conducting said signal to a routing line coupled to a fourth level of interconnections, wherein the span for each of the line and the number of lines of the said fourth level of interconnections are proportional to the said third level of interconnections.

65. The method of Claim 55 further comprising the step of programmably conducting said signal directly to said second logic cluster and bypassing said first set of bi-directionally programmable access lines.

66. The method of Claim 55 further comprising the step of programmably conducting said signal directly to said

second logic block and bypassing said first level of interconnections.

67. The method of Claim 55 further comprising the step of programmably conducting said signal to a first set of bi-directionally programmable access lines of a fifth logic block via a fifth level of interconnections via said fourth level of interconnections by conducting said signal through said first level of interconnections, said second level of interconnections, or said third level of interconnections.

68. The method of Claim 55 further comprising the step of programmably conducting said signal to a first set of bi-directionally programmable access lines of a fifth logic block via a fifth level of interconnections via said fourth level of interconnections without conducting said signal through said first level of interconnections, said second level of interconnections, or said third level of interconnections.

69. The method of Claim 68, wherein the span of each of the said fourth level of interconnections lines is a fraction of the span of each of the said fifth level of interconnections lines and the number of lines of the said fifth level of interconnections lines is a fraction of the number of the said

fourth level of interconnections lines over a logic block circuit area.

70. The method of Claim 55 further comprising the step of programmably conducting said signal to a first set of bi-directionally programmable access lines of a sixth logic block via a sixth level of interconnections through said fourth level of interconnections by conducting said signal through said first level of interconnections, said second level of interconnections, said third level of interconnections, or said fifth level of interconnections.

71. The method of Claim 55 further comprising the step of programmably conducting said signal to a first set of bi-directionally programmable access lines of a sixth logic block via a sixth level of interconnections through said fourth level of interconnections without conducting said signal through said first level of interconnections, said second level of interconnections, said third level of interconnections, or said fifth level of interconnections.

72. The method of Claim 71, wherein the span of each of the said fifth level of interconnections lines is a fraction of the span of each of the said sixth level of interconnections lines and the number of lines of the said sixth level of

interconnections lines is a fraction of the number of the said fifth level of interconnections lines over a logic block circuit area.

73. The method of Claim 55 further comprising the step of selectively coupling an adjacent set of said first set of bi-directionally programmable access lines of an adjacent set of said logic blocks to said first level of interconnections via a set of programmable switches.

74. The method of Claim 55 further comprising the step of selectively coupling an adjacent set of said first set of bi-directionally programmable access lines of an adjacent set of said logic blocks to said second level of interconnections via a set of programmable switches.

75. The method of Claim 55 further comprising the step of selectively coupling an adjacent set of said first set of bi-directionally programmable access lines of an adjacent set of said logic blocks to said third level of interconnections via a set of programmable switches.

76. The method of Claim 55 further comprising the step of selectively coupling an adjacent set of said first set of bi-directionally programmable access lines of an adjacent set

of said logic blocks to said fourth level of interconnections via a set of programmable switches.

77. The method of Claim 55 further comprising the step of selectively coupling an adjacent set of said fourth level of interconnections to said fifth level of interconnections via a set of programmable switches.

78. The method of Claim 55 further comprising the step of selectively coupling an adjacent set of said fourth level of interconnections to said sixth level of interconnections via a set of programmable switches.

79. The method of Claim 55, wherein said switches are comprised of bi-directionally programmable drivers.

80. The method of Claim 55, wherein said switches are comprised of bi-directionally programmable passgates.

81. A field programmable gate array comprising:

a plurality of cells for performing logic functions on signals input to said field programmable gate array;

an intraconnection matrix for programmably interconnecting a plurality of said cells to form a logic cluster;

a plurality of programmable switches connectable between two adjacent said intraconnection matrices to form logic cluster extensions;

a plurality of block connectors together with said logic clusters, said intraconnection matrices, and said extensions to form a logic block;

a plurality of programmable switches connectable between two adjacent said logic blocks to form logic block extensions;

a first level of programmable interconnections for interconnecting a plurality of logic blocks to form a block cluster;

a second level of programmable interconnections for interconnecting a plurality of block clusters to form a block sector;

a first set of programmable switches connecting a plurality of logic blocks to said first level of programmable interconnections;

a second set of programmable switches connecting a plurality of logic blocks to said second level of programmable interconnections;

82. The field programmable gate array of Claim 81, wherein said logic cluster is comprised of a 2 x 2 matrix of cells.

83. The field programmable gate array of Claim 82, wherein said logic block is comprised of a 2 x 2 matrix of logic clusters.

84. The field programmable gate array of Claim 83, wherein said block cluster is comprised of a 2 x 2 matrix of logic blocks.

85. The field programmable gate array of Claim 84, wherein said block sector is comprised of a 2 x 2 matrix of block clusters.

86. The field programmable gate array of Claim 81, wherein said first level of interconnections is comprised of a first set of routing lines and a second set of routing lines perpendicular to said first set of routing lines.

FIG. 1

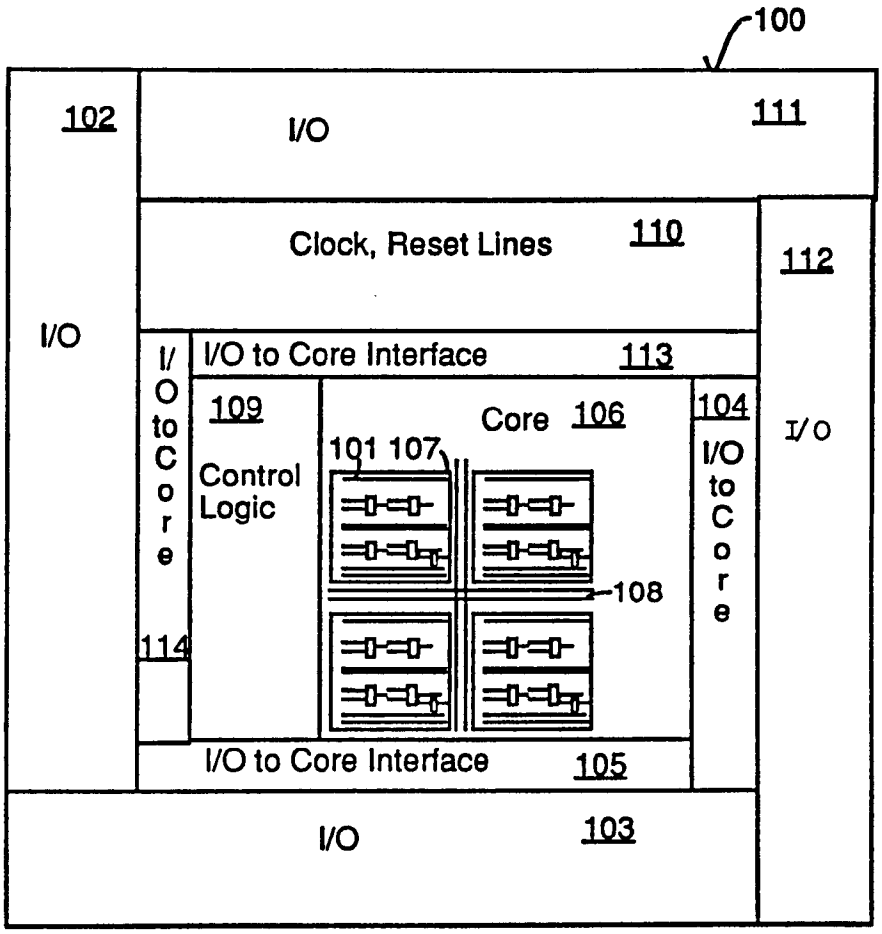

100
102
I/O
111
Clock, Reset Lines
110
112
I/O
I/O to Core Interface
113
I/O to Core
114
109
Control Logic
Core
106
101 107
108
104
I/O to Core
I/O
I/O to Core Interface
105
I/O
103

FIG. 2A

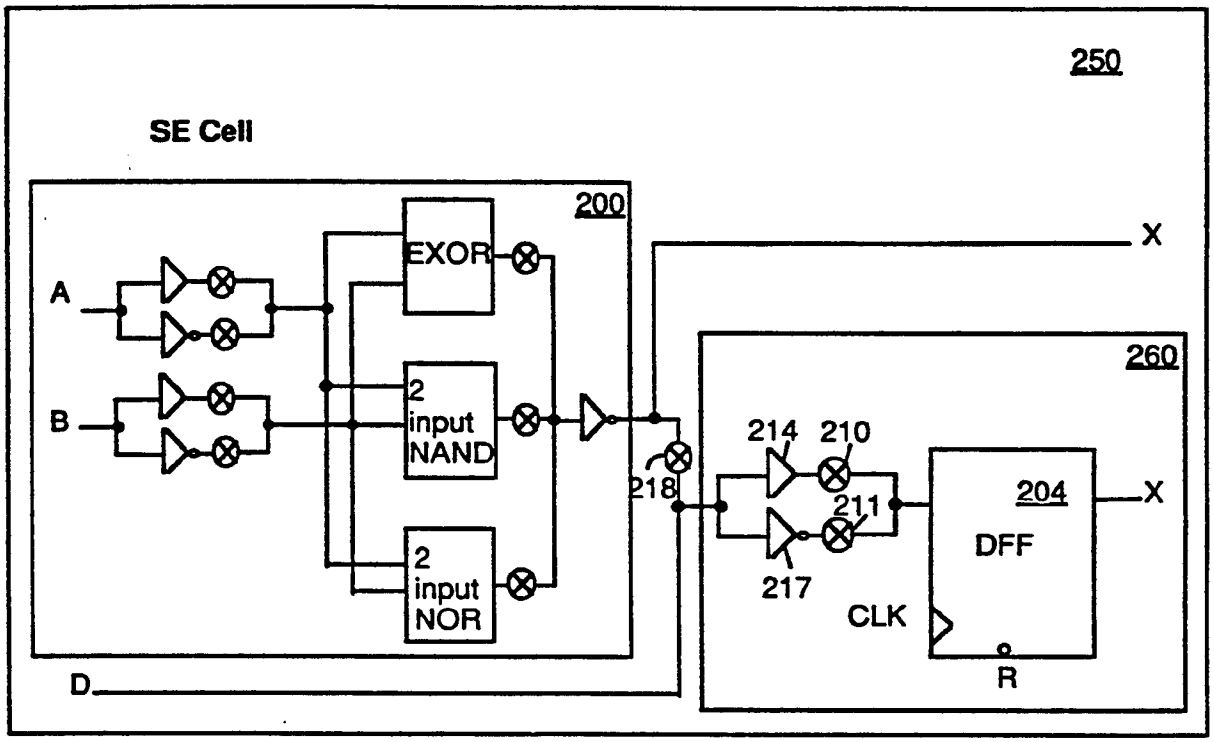

250
SE Cell
200
EXOR
A
B
2 input NAND
2 input NOR
X
260
214
210
211
217
218
204
DFF
CLK
R
D

FIG. 2B

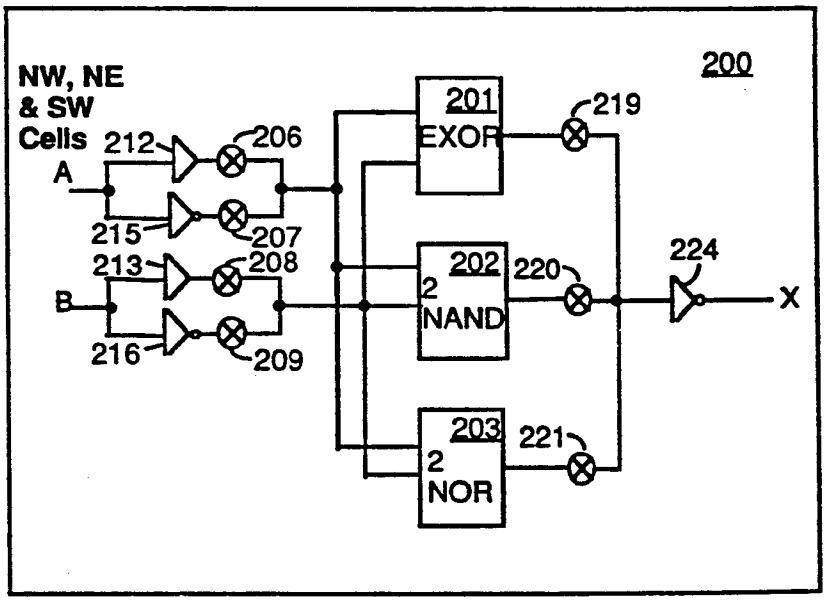

NW, NE & SW Cells
200
201
EXOR
212
206
A
215
207
213
208
B
216
209
202
2 NAND
219
220
224
X
203
2 NOR
221

FIG. 3A

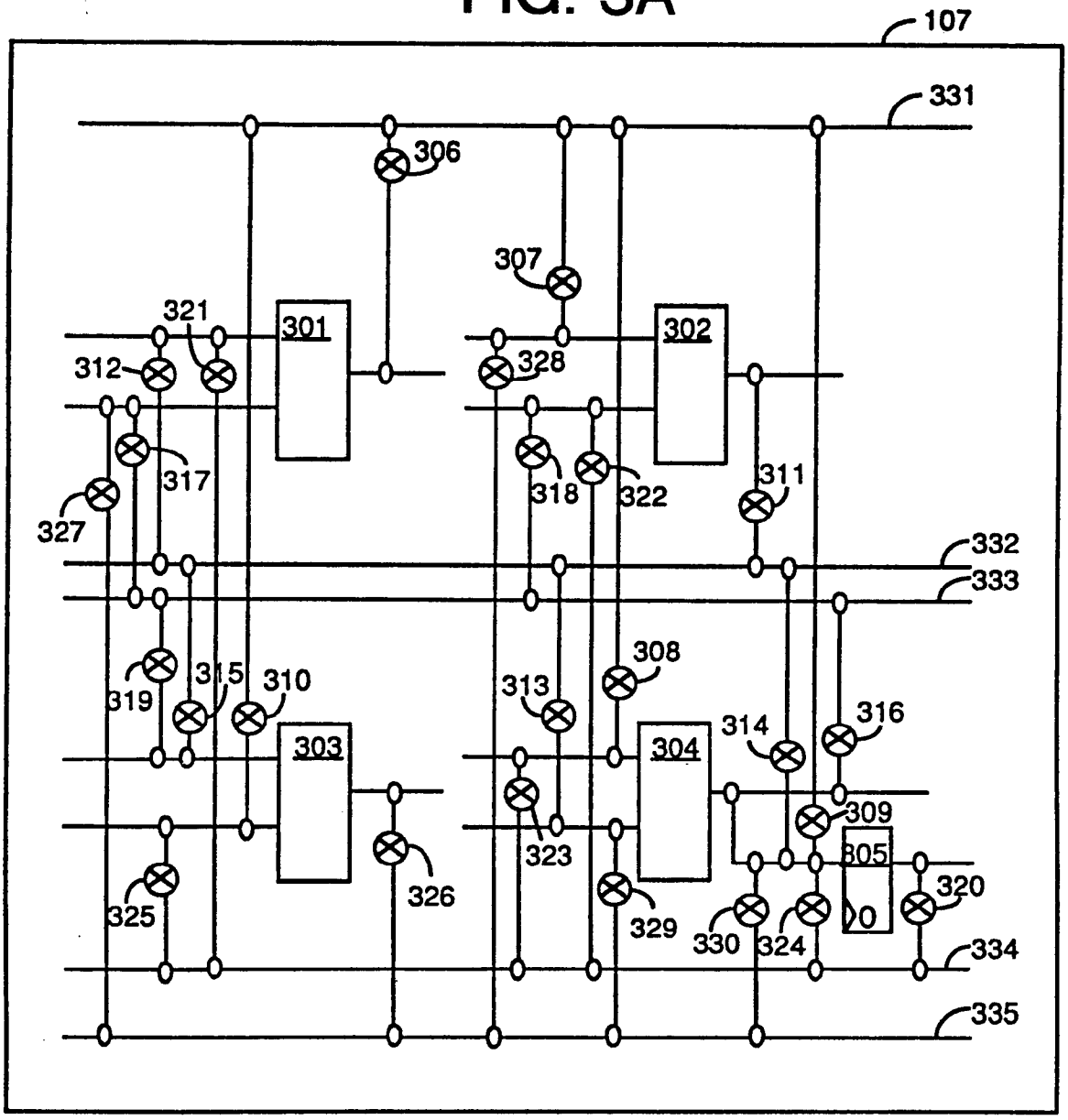

107
331
306
307
321
301
302
312
328
317
318
322
311
327
332
333
308
315
310
313
319
316
314
303
304
309
323
326
325
329
330
324
320
334
335

FIG. 3B

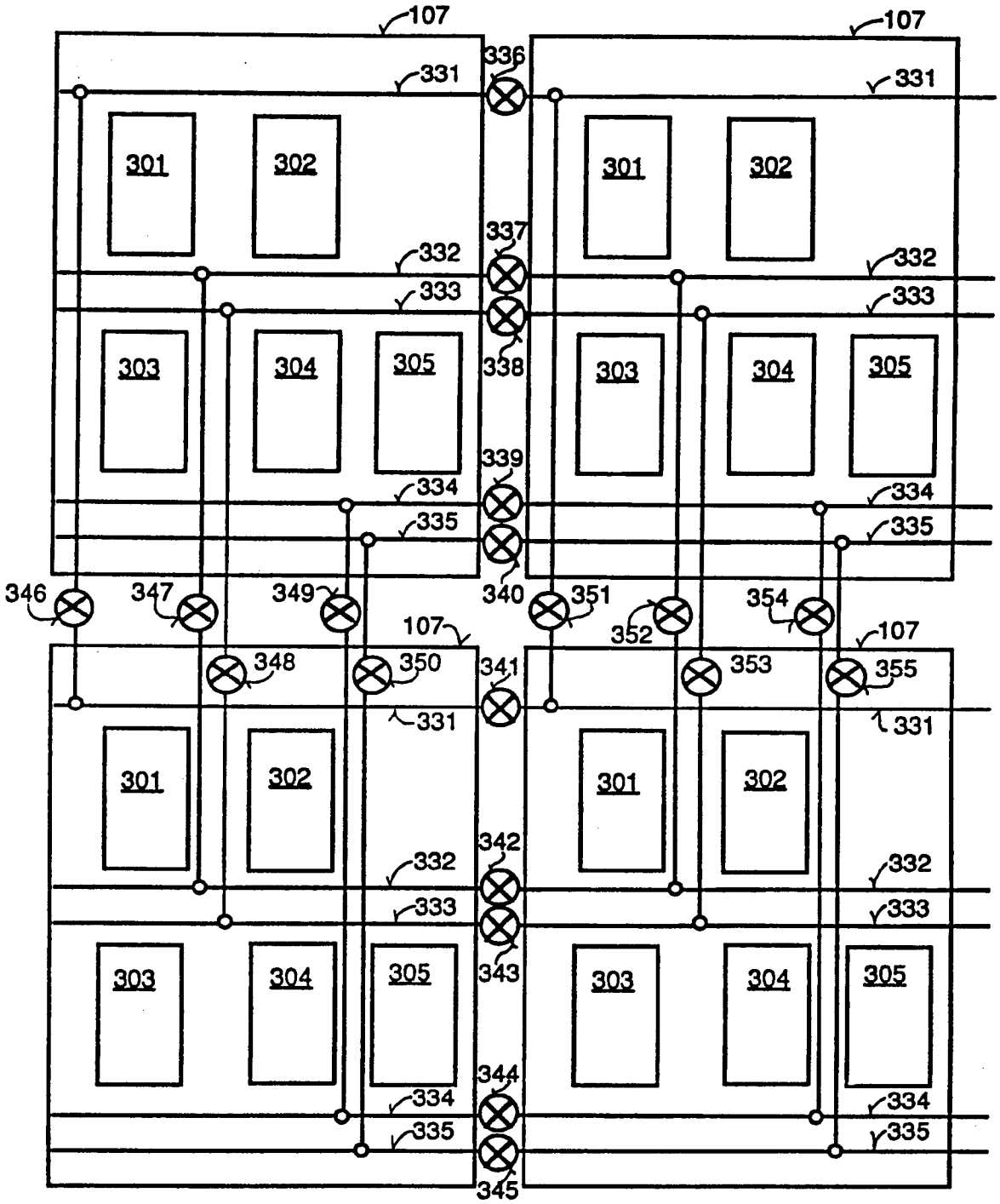

107
331
336
301
302
337
332
333
303
304
305
338
339
334
335
340
346
347
349
351
352
354
348
350
341
353
355
342
343
344
345

FIG. 4A

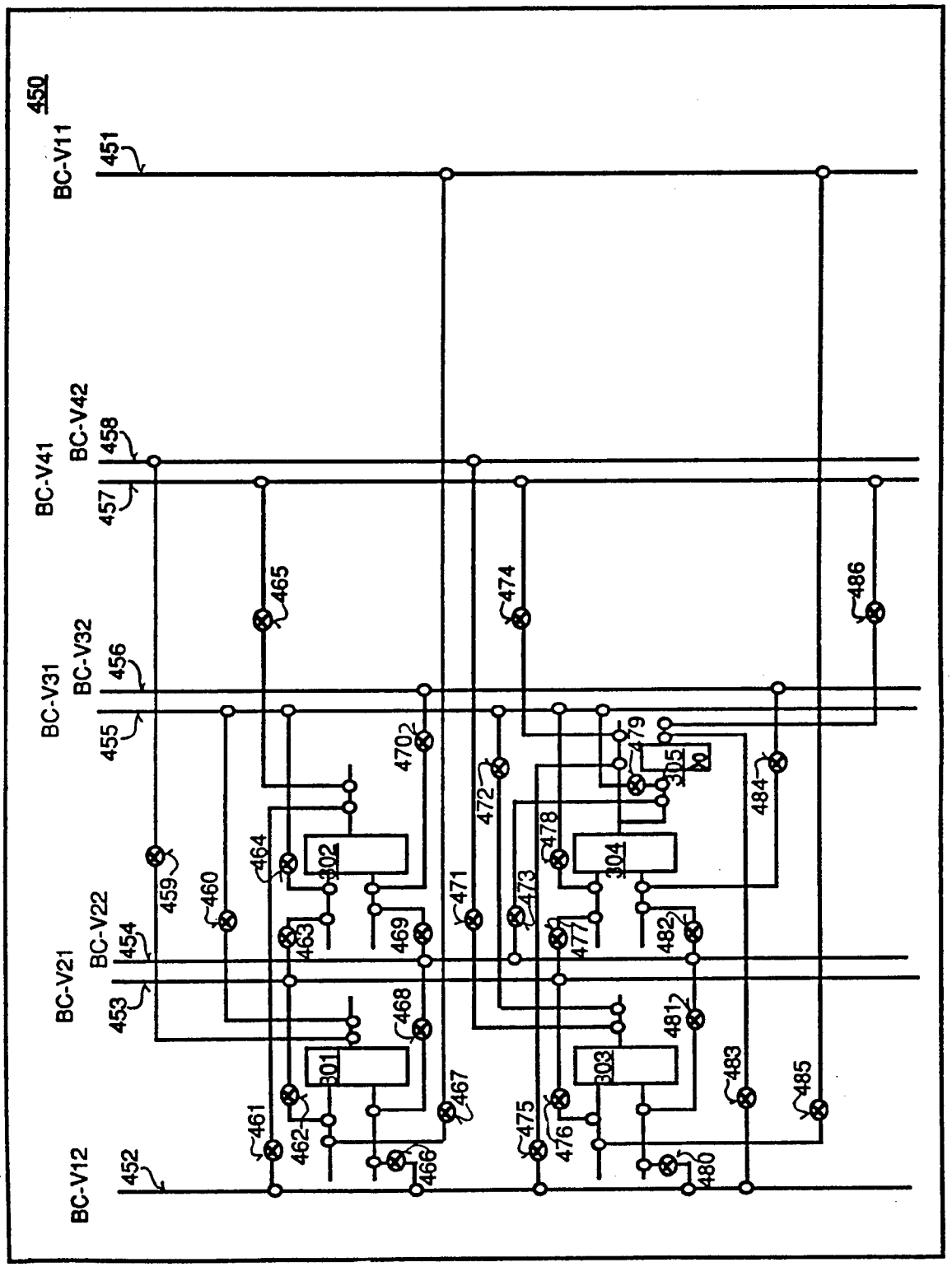

450
BC-V11
451
BC-V42
458
BC-V41
457
BC-V32
456
BC-V31
455
BC-V22
454
BC-V21
453
BC-V12
452
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
301
302
303
304
305

FIG. 4B

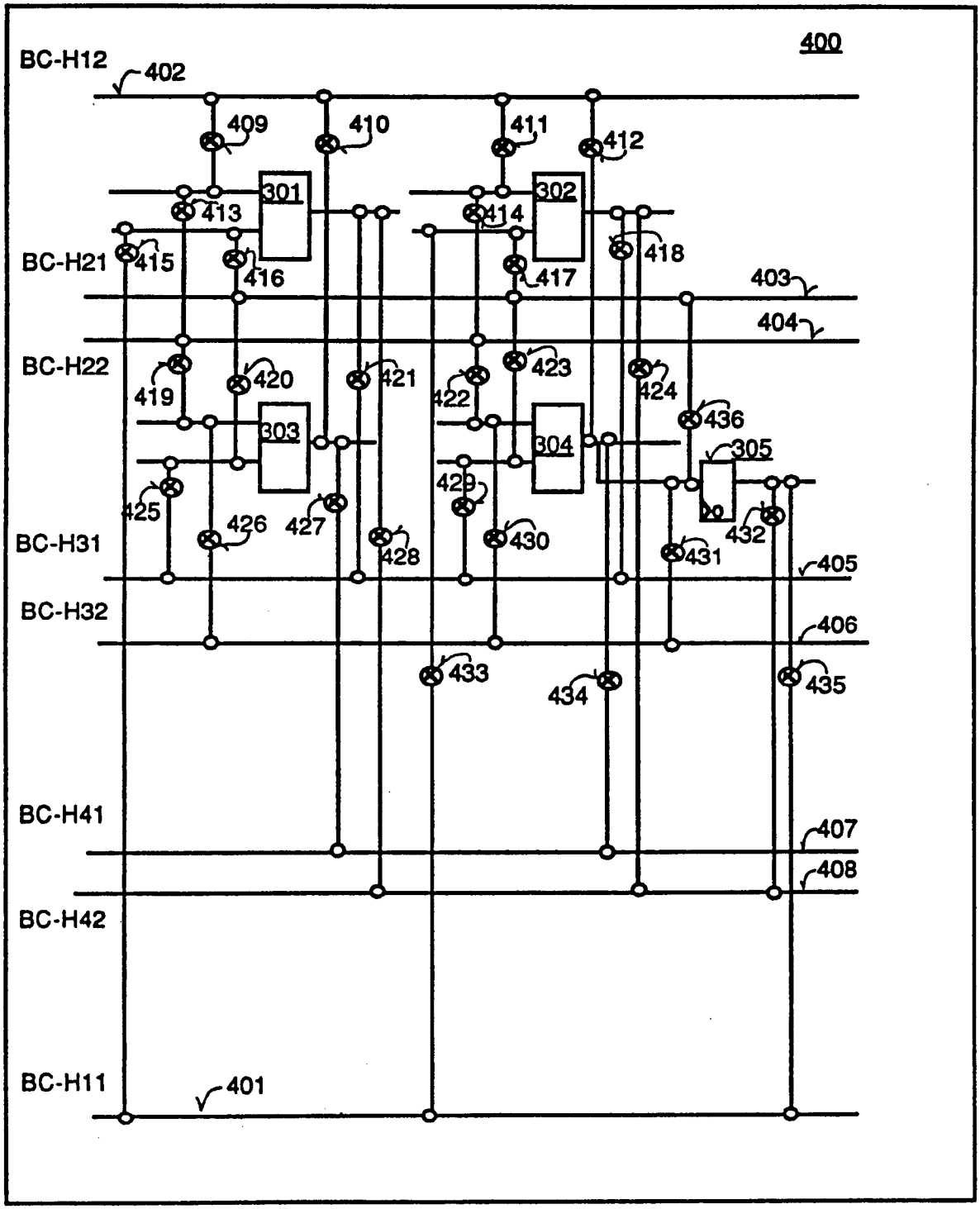

400
BC-H12
402
409
410
411
412
301
302
413
414
BC-H21
415
416
417
418
403
404
BC-H22
419
420
421
422
423
424
303
304
436
305
425
426
427
428
429
430
431
432
BC-H31
405
BC-H32
406
433
434
435
BC-H41
407
408
BC-H42
BC-H11
401

FIG. 5A

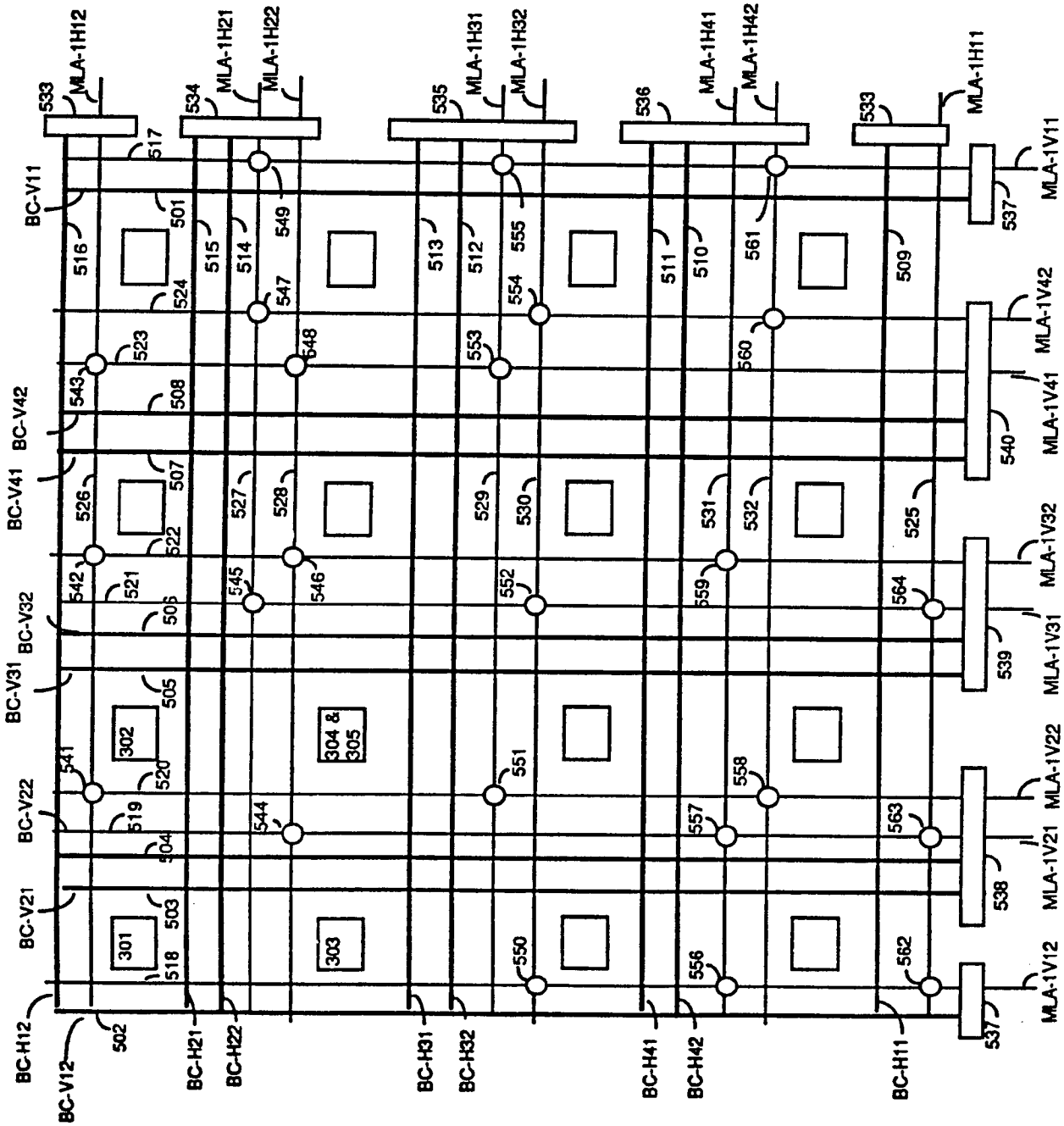

MLA-1H12
MLA-1H21
MLA-1H22
MLA-1H31
MLA-1H32
MLA-1H41
MLA-1H42
MLA-1H11
MLA-1V11
MLA-1V42
MLA-1V41
MLA-1V32
MLA-1V31
MLA-1V22
MLA-1V21
MLA-1V12
BC-V11
BC-V42
BC-V41
BC-V32
BC-V31
BC-V22
BC-V21
BC-V12
BC-H12
BC-H21
BC-H22
BC-H31
BC-H32
BC-H41
BC-H42
BC-H11
301
302
303
304 & 305

FIG. 5B

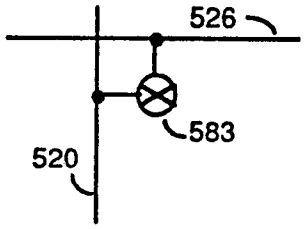

526
583
520

FIG. 5C

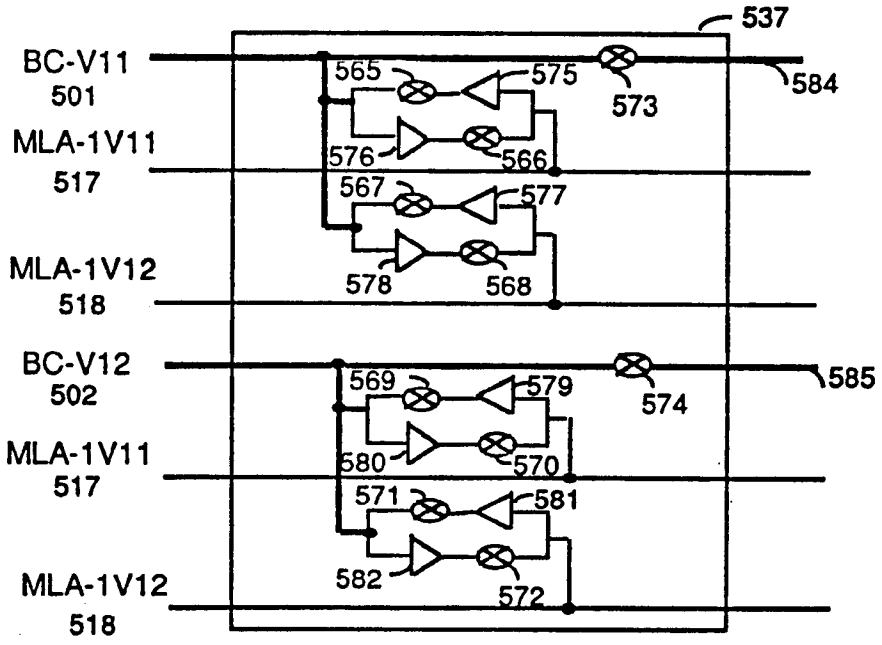

537
BC-V11
501
MLA-1V11
517
MLA-1V12
518
BC-V12
502
MLA-1V11
517
MLA-1V12
518
565
575
573
584
576
566
567
577
578
568
569
579
574
585
580
570
571
581
582
572

FIG. 6

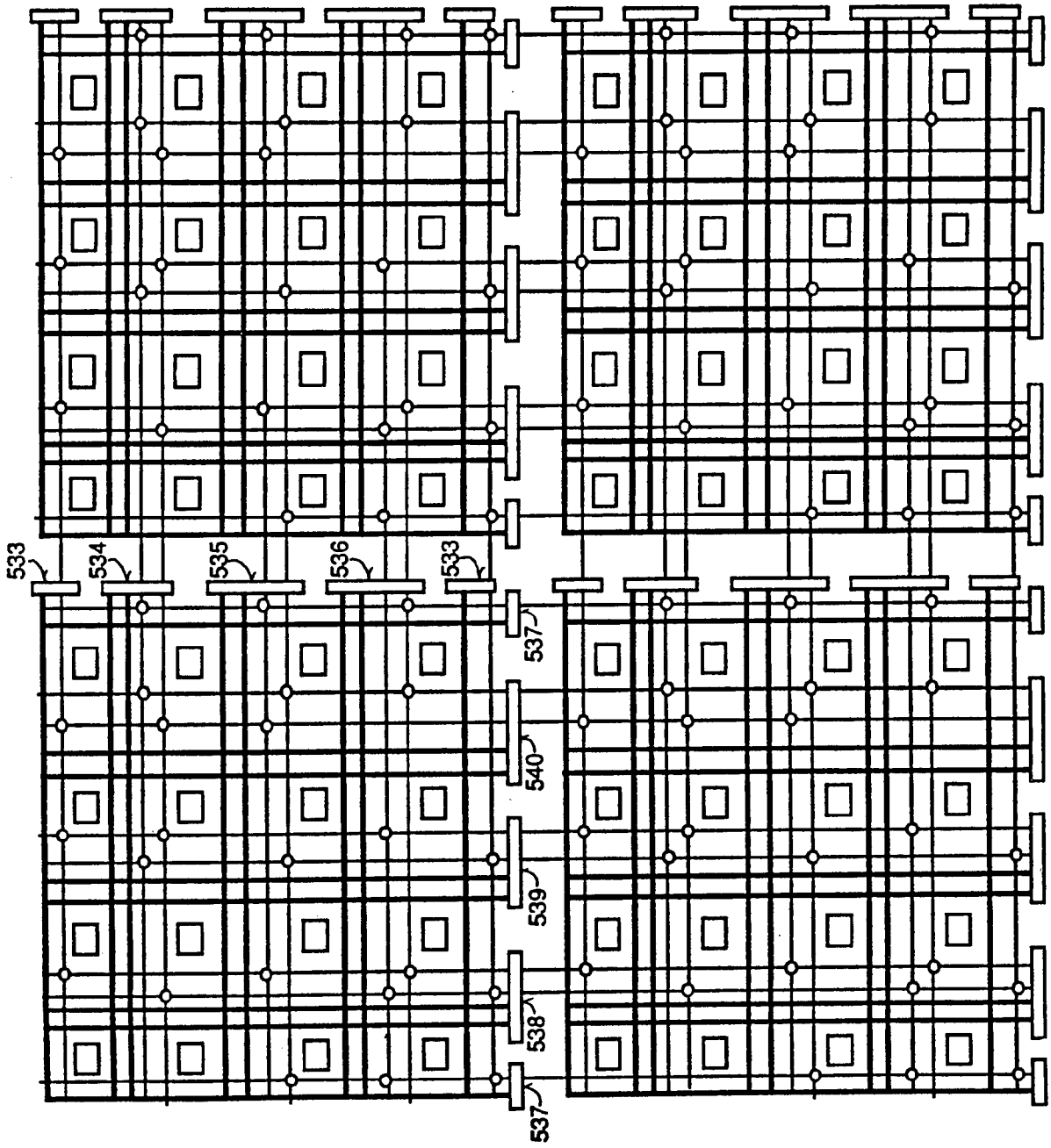


FIG. 7A

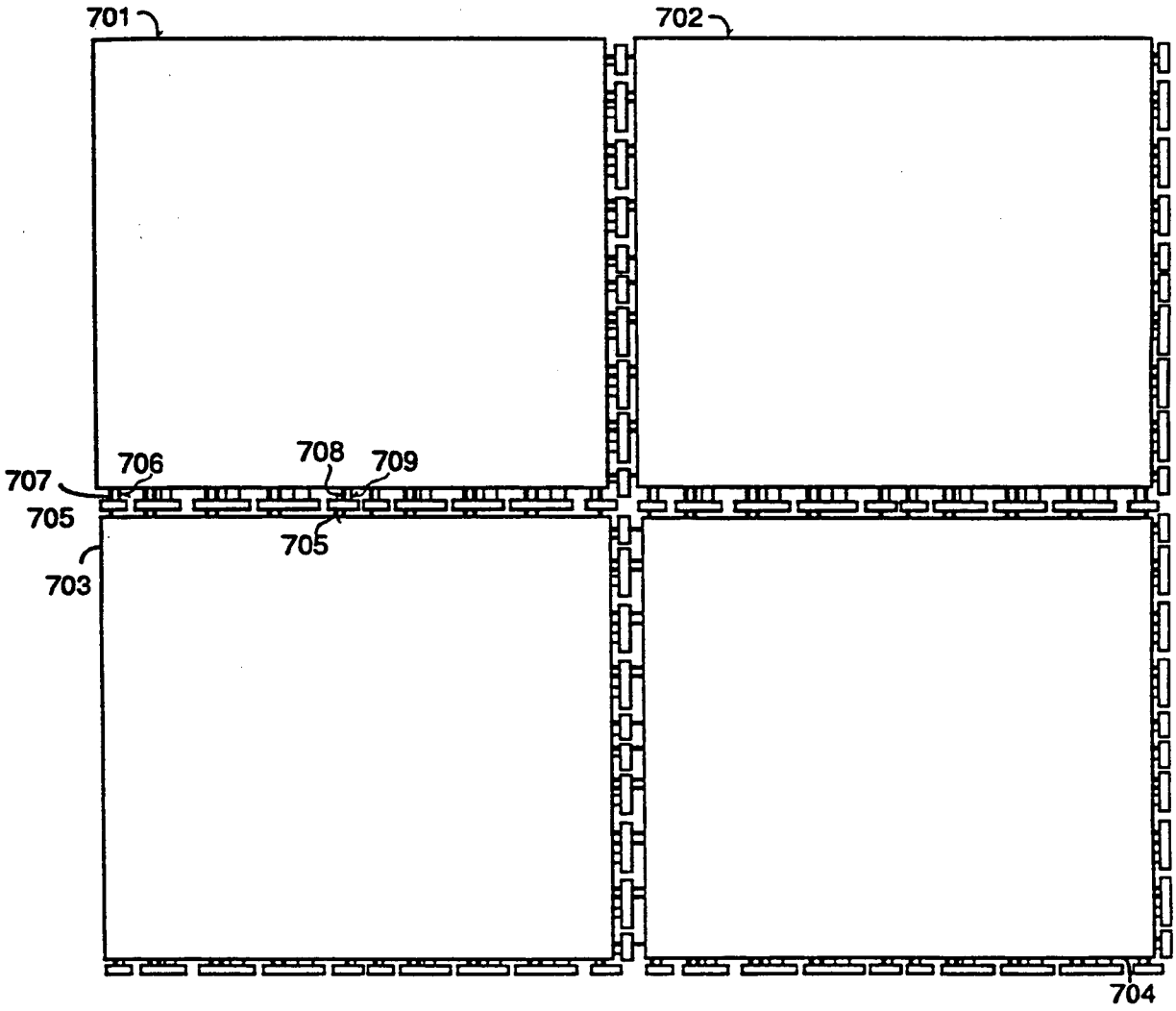

701
702
706
708
709
707
705
705
703
704

FIG. 7B

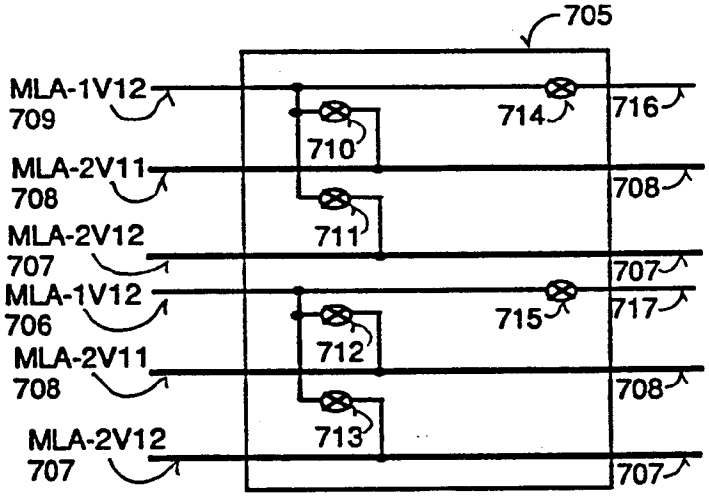

705
MLA-1V12
709
714
716
710
MLA-2V11
708
708
711
MLA-2V12
707
707
MLA-1V12
706
715
717
712
MLA-2V11
708
708
713
MLA-2V12
707
707

FIG. 8A

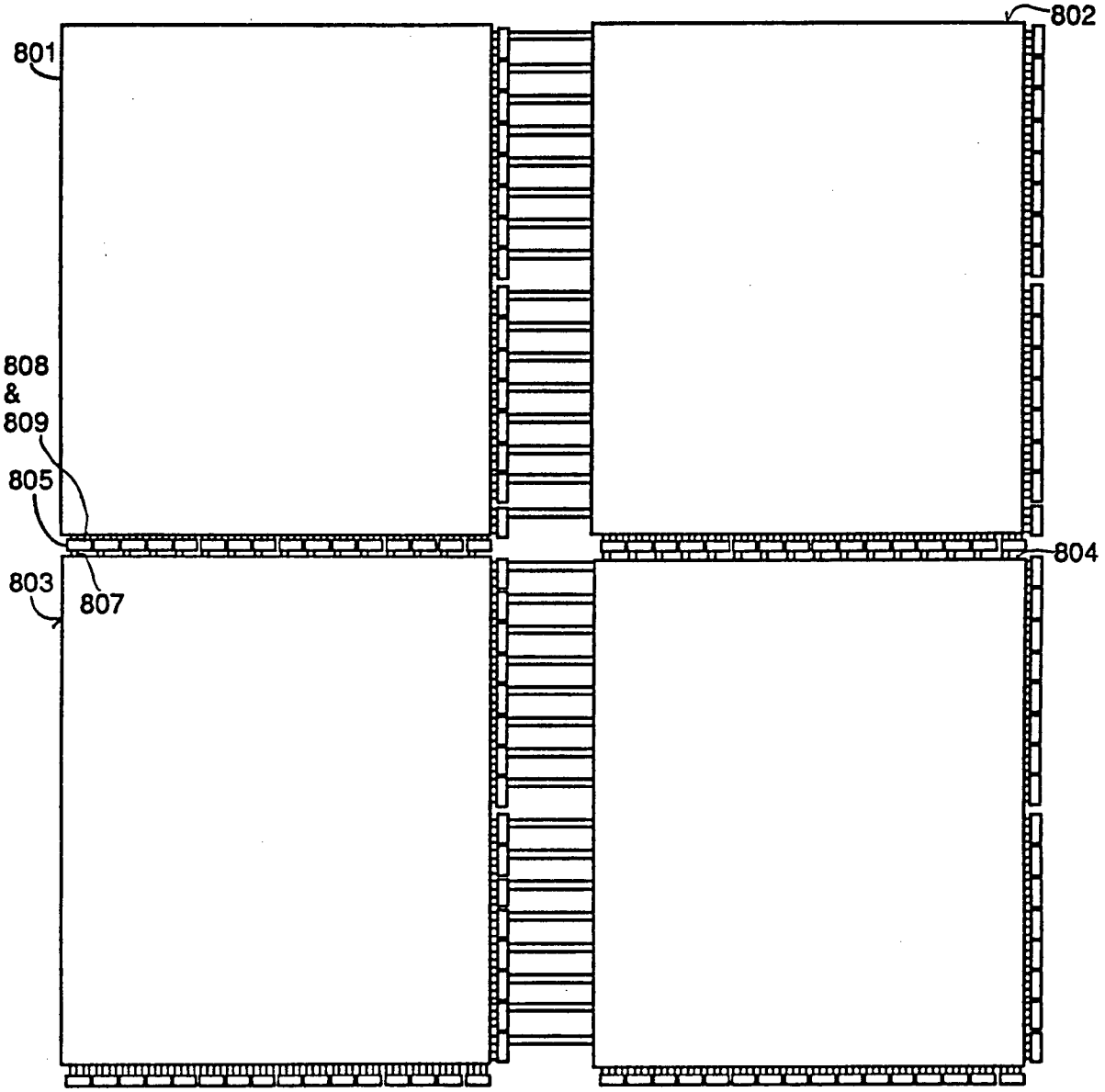

801
802
808
&
809
805
804
803
807

FIG. 8B

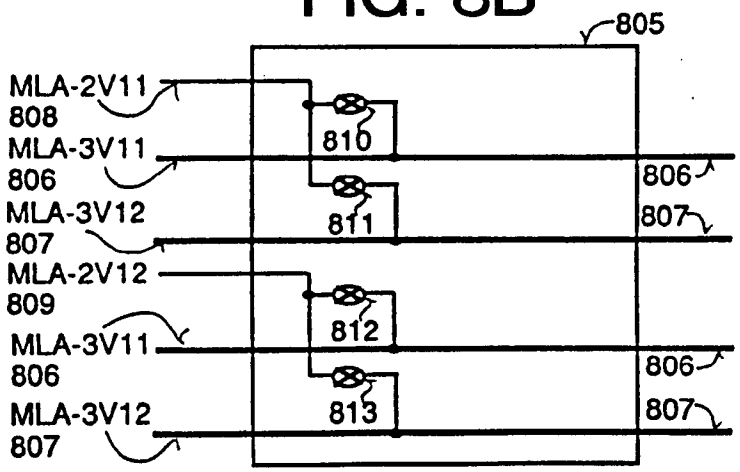

805
MLA-2V11
808
MLA-3V11
806
MLA-3V12
807
MLA-2V12
809
MLA-3V11
806
MLA-3V12
807
810
811
812
813
806
807
806
807

FIG. 9

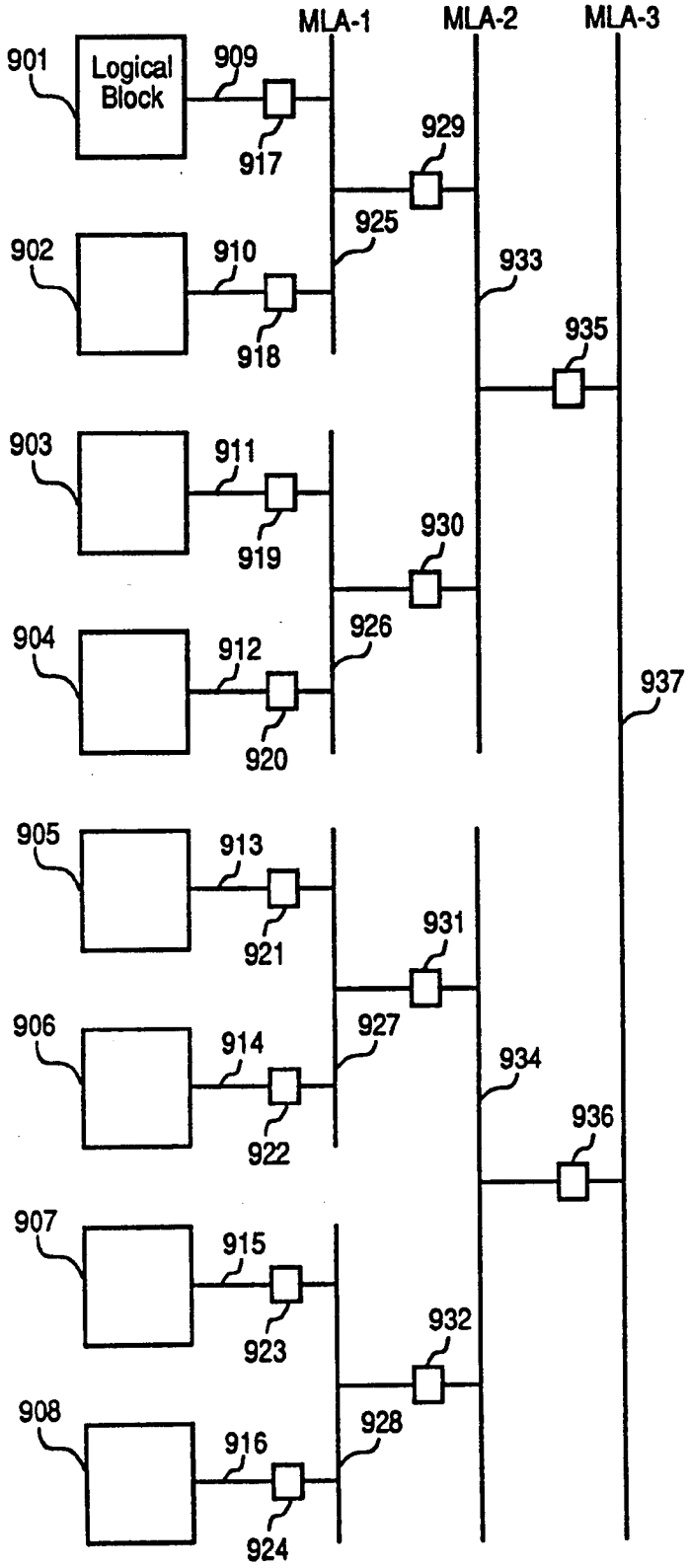

MLA-1
MLA-2
MLA-3
901
Logical Block
909
917
929
925
902
910
918
933
935
903
911
919
930
926
904
912
920
937
905
913
921
931
906
914
922
927
934
936
907
915
923
932
908
916
924
928

FIG. 10

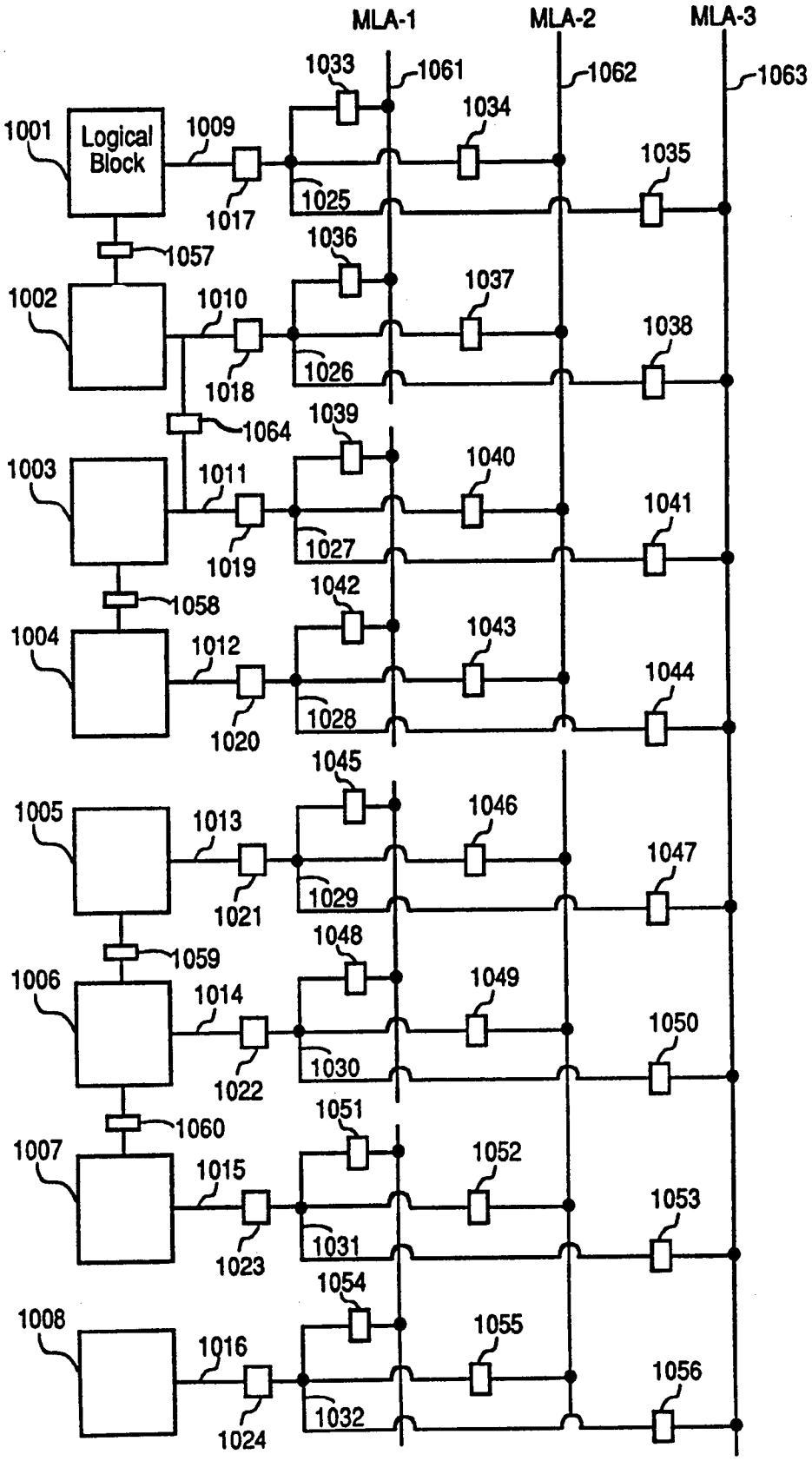

MLA-1
MLA-2
MLA-3
1001
Logical Block
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064

FIG. 11

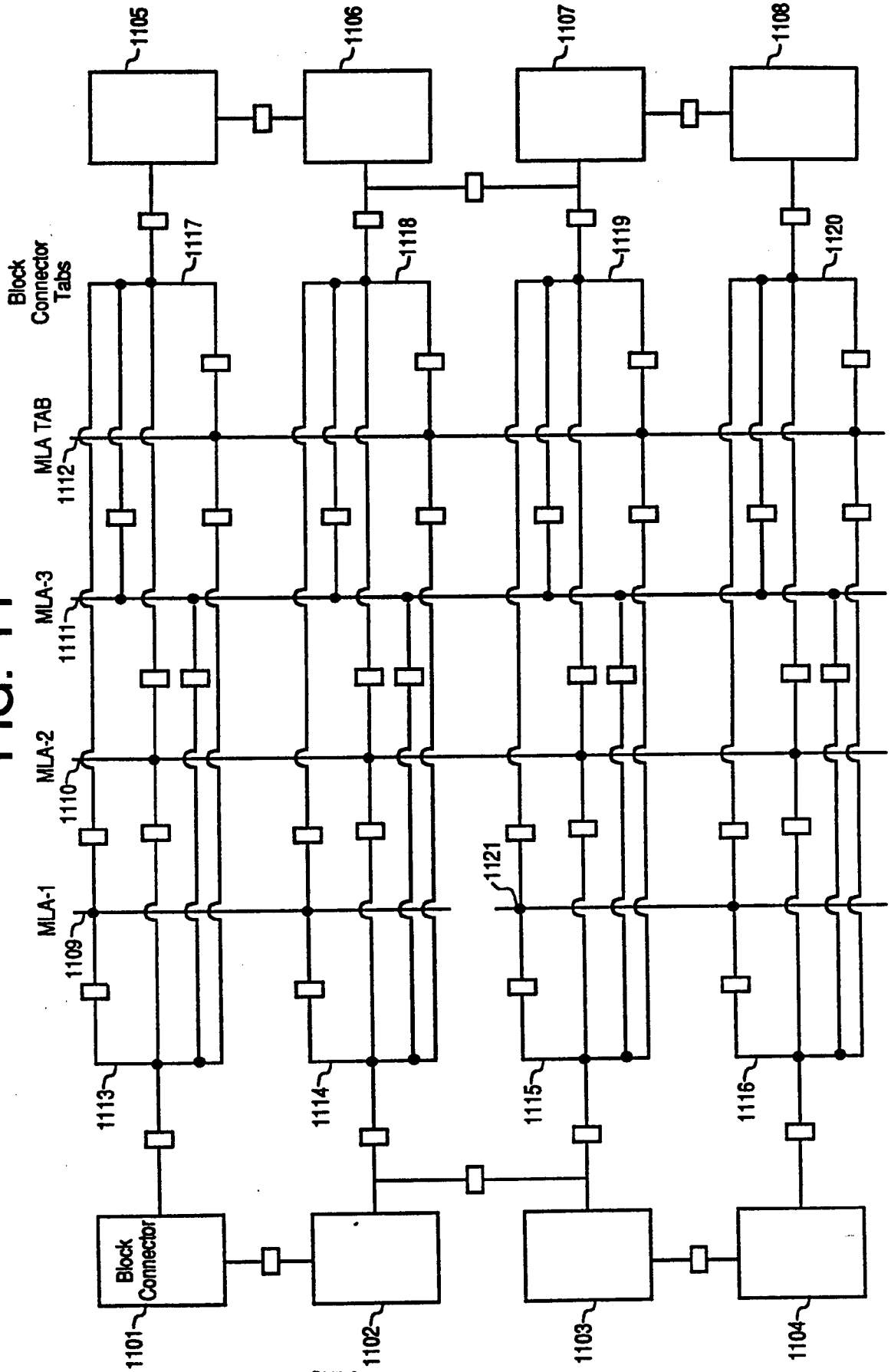

1105
1106
1107
1108
Block Connector Tabs
1117
1118
1119
1120
MLA TAB
1112
MLA-3
1111
MLA-2
1110
MLA-1
1109
1121
1113
1114
1115
1116
Block Connector
1101
1102
1103
1104

FIG. 12A

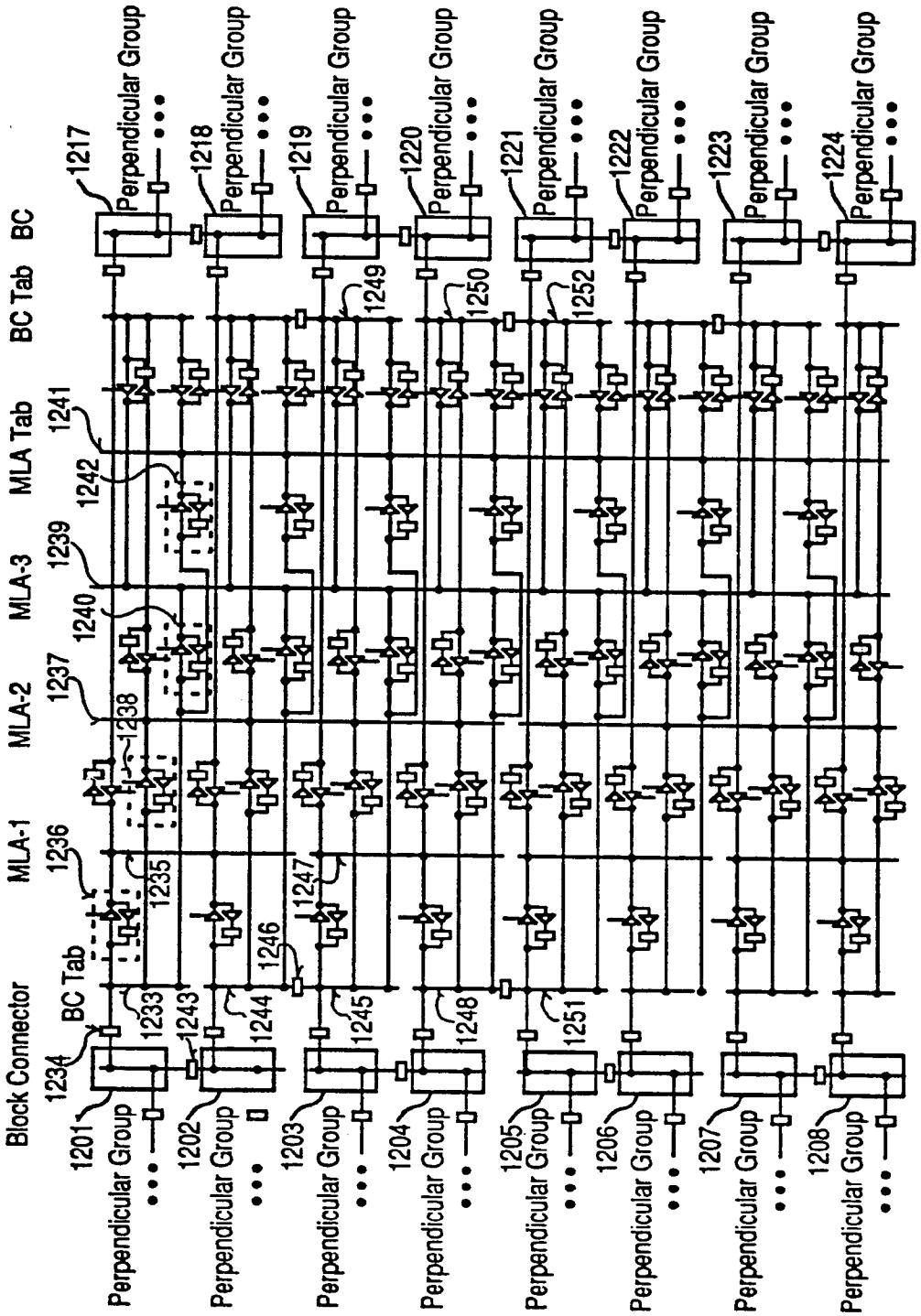

Block Connector
BC Tab
MLA-1
MLA-2
MLA-3
MLA Tab
BC Tab
BC
Perpendicular Group
1201
1202
1203
1204
1205
1206
1207
1208
1217
1218
1219
1220
1221
1222
1223
1224
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252

FIG. 12B

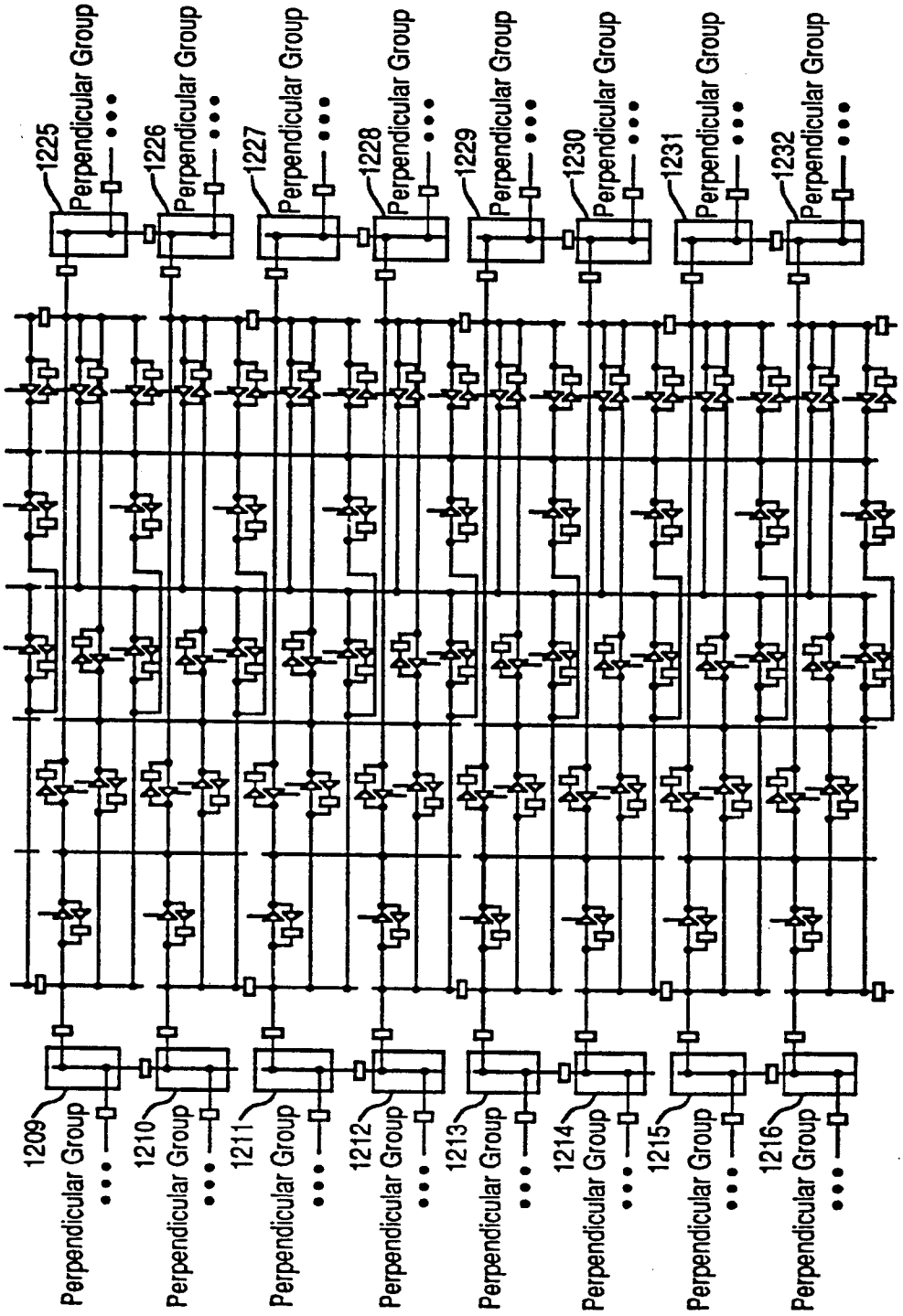

1225
Perpendicular Group
1226
Perpendicular Group
1227
Perpendicular Group
1228
Perpendicular Group
1229
Perpendicular Group
1230
Perpendicular Group
1231
Perpendicular Group
1232
Perpendicular Group
1209
Perpendicular Group
1210
Perpendicular Group
1211
Perpendicular Group
1212
Perpendicular Group
1213
Perpendicular Group
1214
Perpendicular Group
1215
Perpendicular Group
1216
Perpendicular Group

FIG. 13

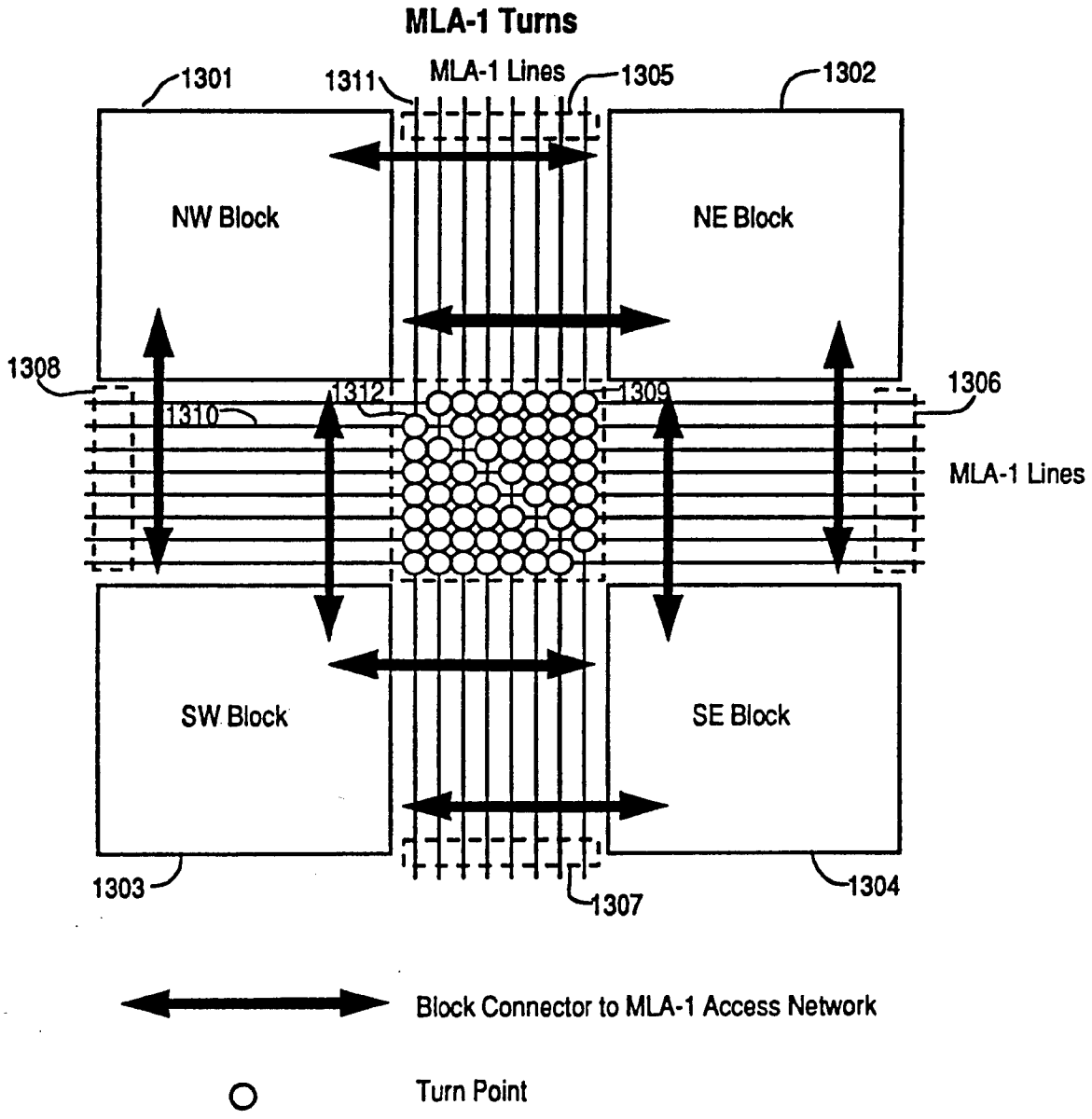

MLA-1 Turns
1301
1311
MLA-1 Lines
1305
1302
NW Block
NE Block
1308
1312
1309
1306
1310
MLA-1 Lines
SW Block
SE Block
1303
1307
1304
Block Connector to MLA-1 Access Network
Turn Point

FIG. 14

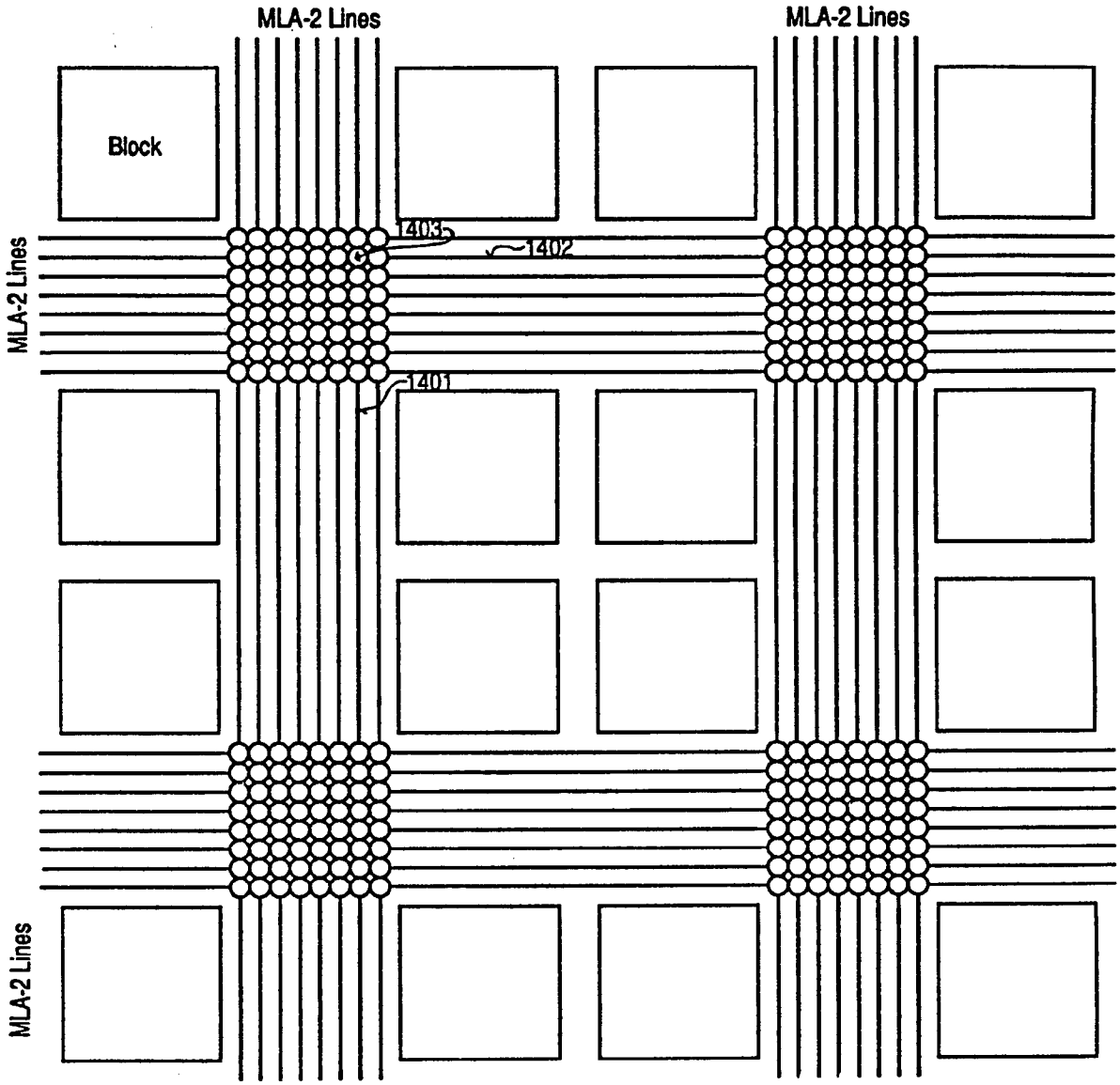

MLA-2 Lines
MLA-2 Lines
Block
1403
1402
MLA-2 Lines
1401
MLA-2 Lines
Turn Point

FIG. 15

MLA-3 Turn

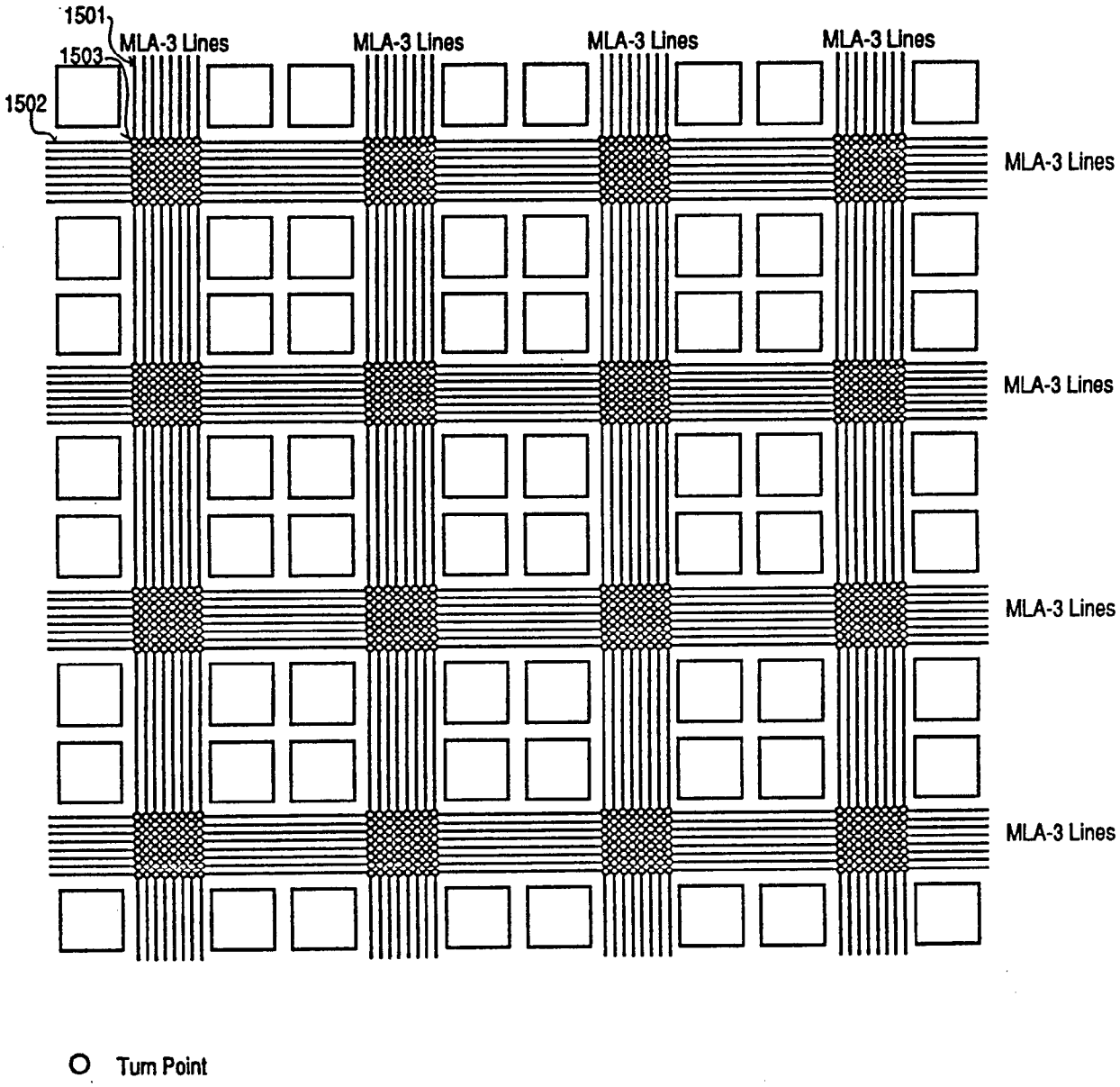
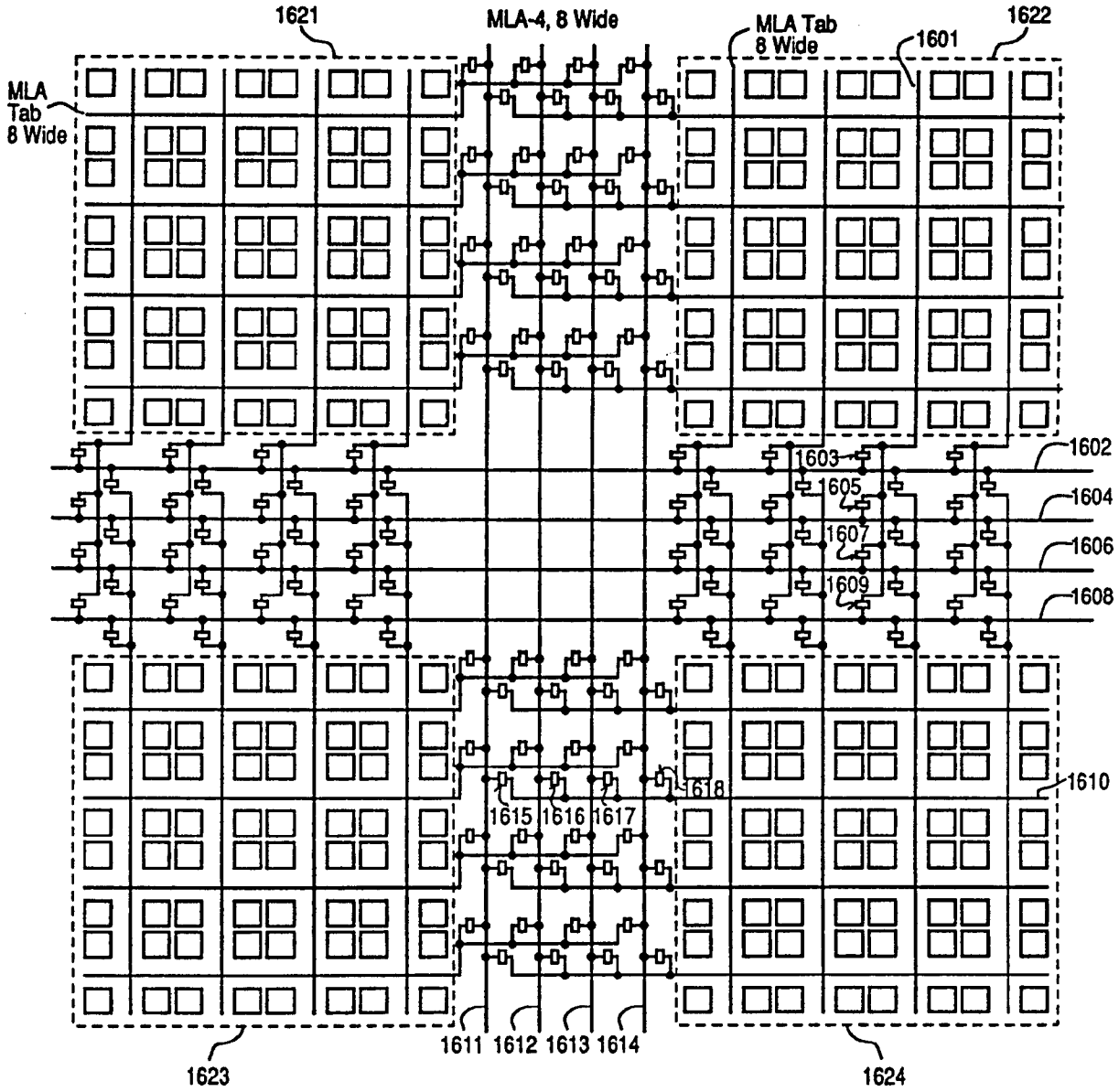


FIG. 16

MLA-4 Routing Network - Access and Routing



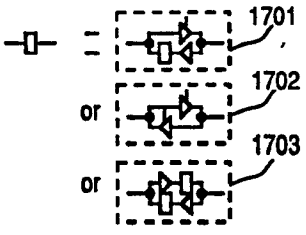
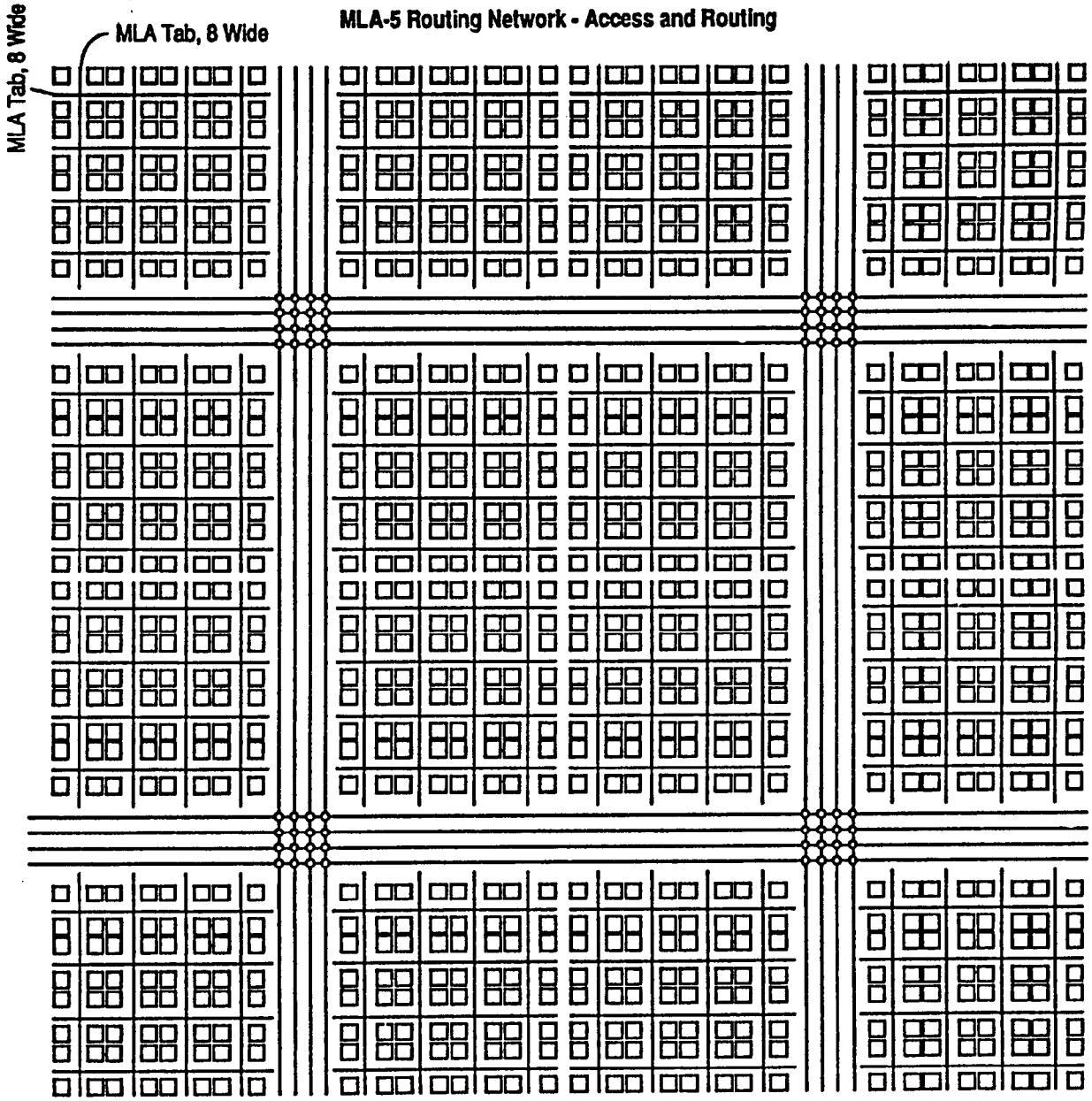


FIG. 17

FIG. 18


MLA-5 Routing Network - Access and Routing
MLA Tab, 8 Wide
MLA Tab, 8 Wide

INTERNATIONAL SEARCH REPORT

Internati . Application No
PCT/US 95/04639

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 H03K19/177

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 H03K

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	ELECTRONIC DESIGN, vol. 41, no. 20, 1 October 1993 CLEVELAND, OH, US, pages 33-34, XP 000400299 D. BURSKY 'Fine - Grain FPGA Architecture Uses Four Levels of Configuration Hierarchy' see the whole document ---	1-4, 42-48, 55-58, 81-86
X	PROCEEDINGS OF THE IEEE 1993 CUSTOM INTEGRATED CIRCUITS CONFERENCE, 9 May 1993 NEW YORK, US, pages 7.3.1-7.3.5, XP 000409658 R. CLIFF ET AL. 'A Dual Granularity and Globally Interconnected Architecture for a Programmable Logic Device' see the whole document ---	1-4, 42-48, 55-58,81

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
28 June 1995	07. 07. 95

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Blaas, D-L

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Internat. Application No
PCT/US 95/04639

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
P,X	WO,A,94 10754 (XILINX INC) 11 May 1994 see page 8, line 11 - page 9, line 33; figures 1,2,4,5,7,20	1-4, 42-48, 55-58, 81-86
P,X	EP,A,0 630 115 (PILKINGTON MICRO ELECTRONICS) 21 December 1994 see page 9, line 32 - line 39; figure 10	1-4, 42-48, 55-58, 81-86

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Information on patent family members

Internat. Application No
PCT/US 95/04639

Patent document cited in search report	Publication date	Patent family member(s)		Publication date
WO-A-9410754	11-05-94	NONE		
EP-A-0630115	21-12-94	AU-B-	6465494	22-12-94
		CA-A-	2125307	19-12-94
		GB-A-	2279168	21-12-94
		JP-A-	7058631	03-03-95